

# **ACTIVE SUPPORT FOR INSTRUCTORS AND STUDENTS IN AN ONLINE LEARNING ENVIRONMENT**

A Thesis Submitted to the College of

Graduate Studies and Research

in Partial Fulfillment of the Requirements

for the Degree of Master of Science

in the Department of Computer Science

University of Saskatchewan

Saskatoon

By

Collene Hansen

Keywords: learner model, open learner model, active learner model, intelligent tutoring system, learner content management system

© Copyright Collene Hansen, August 2007. All rights reserved.

## PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in this thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or in part should be addressed to:

Head of the Department of Computer Science

University of Saskatchewan

Saskatoon, Saskatchewan, Canada

S7N 5A9

## ABSTRACT

By opening the learner model to both the learner and other peers within an e-learning system, the learner gains control over his or her learner model and is able to reflect on the contents presented in the model. Many current modeling systems translate an existing model to fit the context when information is needed. This thesis explores the observation that information in the model depends on the context in which it is generated and describes a method of generating the model for the specific user and purpose. The main advantage of this approach is that exactly the right information is generated to suit the context and needs of the learner. To explore the benefits and possible downsides of this approach, a learner model Query Tool was implemented to give instructors and learners the opportunity to ask specific questions (queries) of the content delivery system hosting several online courses. Information is computed in real time when the query is run by the instructor, so the data is always up-to-date. Instructors may then choose to allow students to run the query as well, enabling learner reflection on their progress in the course as the instructor has defined it. I have called this process *active open learner modelling*, referring to the open learner modelling community where learner models are accessible by learners for reflective purposes, and referring to the active learner modelling community which describes learner modelling as a context-driven process. Specific research questions explored in this thesis include “how does context affect the modelling process when learner models are opened to users”, “how can privacy be maintained while useful information is provided”, and “can an accurate and useful learner model be computed actively”.

## ACKNOWLEDGEMENTS

I would like to thank my supervisor Dr. Gordon McCalla for his support, guidance, patience, and words of wisdom throughout this project. Thank you for the stimulating conversations to open new ideas and interests to me and your optimism during my many times of doubt in myself.

Thank you to my advisor committee: Dr. Jim Greer, Dr. Julita Vassileva, and Dr. Kalyani Premkumar for their valuable suggestions, comments, and future directions for projects extending from this research.

Thank you as well to current and former fellow students of the ARIES Lab and the many members of the iHelp Team for so many valuable discussions and support, especially Christopher Brooks, Lori Kettel, Jian Liu, and Zinan Guo.

Thanks to my current and past fellow staff members of the Computer Science Department for encouragement, technical help, and the occasional well-timed distraction, particularly Sonia Chiasson, Guus van de Velde, and Gina Koehn. I appreciate the support the department has shown as I have completed my degree.

Thanks and love to my family and extended family for your support and interest in my education and career, especially my uncles Ron Hansen and Merv Gunderson.

Finally, a thank you to my parents, Roy and Donna Hansen, for your love and your motivation to do my best in whatever direction I am working towards.

# TABLE OF CONTENTS

	<u>page</u>
PERMISSION TO USE .....	i
ABSTRACT .....	ii
ACKNOWLEDGEMENTS .....	iii
TABLE OF CONTENTS .....	iv
LIST OF TABLES .....	vi
LIST OF FIGURES .....	vii
1. INTRODUCTION .....	1
2. LITERATURE SURVEY .....	4
2.1. <i>User Modelling and Learner Modelling</i> .....	4
2.2. <i>Open Learner Modelling</i> .....	5
2.2.1. Viewing or Scrutiny .....	5
2.2.2. Interaction .....	7
2.2.3. Motivation for Open Modelling.....	8
2.2.3.1. Reflection .....	8
2.2.3.2. Validation.....	9
2.2.3.3. Assessment.....	10
2.2.3.4. Other Motivations .....	10
2.2.4. Visualization of the Open Model .....	10
2.2.5. Summary of Open Learner Modelling .....	11
2.3. <i>Active Learner Modelling</i> .....	12
2.3.1. A Definition of Active Modelling.....	12
2.4. <i>Exploring Open Modelling in an Active Context</i> .....	14
2.4.1. Purposes .....	15
2.4.2. Learners.....	16
2.4.3. Motivation.....	17
2.5. <i>Conclusion</i> .....	19
3. ACTIVE OPEN QUERY TOOL .....	20
3.1. <i>Active Open Learner Modelling Query Tool</i> .....	21
3.1.1. Uses of the Query Tool .....	24
3.1.2. Creating a View.....	25
3.1.2.1. Part 1: Adding Characteristics to the View.....	25
3.1.2.2. Part 2: Filtering Learners in and out of a View.....	28
3.1.2.3. Part 3: Opening the View to Learners by Setting Privileges.....	34

3.1.2.4. Part 4: Describing the View's Purpose .....	35
3.1.3. Privacy Protection .....	35
3.1.4. Listing and Generating Views.....	37
3.1.5. Real Views and Generated Results for Instructors .....	39
3.1.6. Observations from Real Views for Instructors.....	50
3.1.7. Real Views and Sample Results for Students .....	53
3.1.8. Observations from Real Views for Students.....	57
3.1.9. Summary of the Query Tool .....	59
3.2. <i>Research Questions</i> .....	62
4. STUDY AND RESULTS .....	63
4.1. <i>Overview of Study Procedure</i> .....	63
4.2. <i>Instructor Statistics, Survey, and Results</i> .....	64
4.2.1. Query Tool Usage and Usefulness.....	64
4.2.2. Usability of the Query Tool .....	73
4.2.3. Accuracy of the Query Tool.....	78
4.2.4. Effects of the Query Tool on Teaching and Learning.....	79
4.2.5. Privacy and the Query Tool .....	79
4.3. <i>Learner Statistics, Survey, and Results</i> .....	80
4.3.1. Query Tool Usage and Usefulness.....	83
4.3.2. Usability of the Query Tool .....	90
4.3.3. Accuracy of the Query Tool.....	93
4.3.4. Effects of the Query Tool on Learning .....	95
4.3.5. Privacy and the Query Tool .....	100
4.4. <i>Summary and Conclusions from Study</i> .....	105
5. SUMMARY AND CONCLUSION.....	107
5.1. <i>Summary</i> .....	107
5.2. <i>Future Directions</i> .....	110
5.3. <i>Conclusion</i> .....	114
REFERENCES.....	116
APPENDIX A: CHARACTERISTICS AVAILABLE IN QUERY TOOL.....	124
APPENDIX B: FILTERS AVAILABLE IN QUERY TOOL.....	132
APPENDIX C: QUERY TOOL SYSTEM DESIGN .....	135
APPENDIX D: INSTRUCTOR CONSENT FORM AND SURVEY .....	141
APPENDIX E: LEARNER CONSENT FORM AND SURVEY .....	156
APPENDIX F: QUERY TOOL USER'S GUIDE .....	169

## LIST OF TABLES

<u>Table</u>	<u>page</u>
4-1. Summary of Courses using the Query Tool .....	65
4-2. Views created for Course 1 (CMPT 100 regular section) .....	67
4-3. Views created for Course 2 (Transforming Teaching) .....	68
4-4. Views created for Course 3 (CMPT 408) .....	69
4-5. Time taken to generate views in Course 1 (CMPT 100 regular section).....	76
4-6. Time taken to generate views in Course 2 (Transforming Teaching) .....	77
4-7. Time taken to generate views in Course 3 (CMPT 408) .....	78
4-8. Time taken to generate views in Term 1 considering only student usage .....	91
4-9. Time taken to generate views in Term 2 considering only student usage .....	91

## LIST OF FIGURES

<u>Figure</u>	<u>page</u>
3-1. iHelp Courses and the communications bar .....	23
3-2. Predecessor of Query Tool .....	24
3-3. Table view creation interface.....	26
3-4. Adding characteristics to a view .....	26
3-5. Clarifying characteristic “highest score on quiz” with a list of quizzes .....	27
3-6. Clarifying characteristic “number of threads started in category” with list of categories.....	27
3-7. Clarifying characteristic “number of posts read in course” with list of courses and date .....	28
3-8. Default learners to include in the view .....	29
3-9. Filtering results on a view based on who has recently logged on.....	30
3-10. Filtering by role .....	31
3-11. Filtering by page views.....	32
3-12. Filtering by name.....	33
3-13. Filtering in students who have logged on since January 1, 2007 .....	34
3-14. Giving access to student assistants and CMPT 100 in class students to generate the view.....	35
3-15. Setting privileges for characteristics based on roles .....	36
3-16. Compare progress feature in iHelp Courses .....	38
3-17. Created views for a course.....	38
3-18. Contact information for online students who have not yet logged on.....	40
3-19. Which students are currently online .....	41
3-20. When should I log on to see someone online .....	42
3-21. What time of the day do learners log on .....	43
3-22. Is there anyone falling behind.....	44
3-23. Who uses the cd .....	45
3-24. Are the students actively viewing content .....	46
3-25. Participation marks by postings made and started.....	47
3-26. Participation marks by postings made and read .....	49
3-27. Example tree view .....	51
3-28. Example timeline view .....	51
3-29. How am I doing .....	54
3-30. View created to list contact information for guests .....	55
3-31. View generated by instructor.....	56
3-32. View generated by learner .....	56
4-1. Number of times instructors generated views.....	70
4-2. Reasons instructors used the Query Tool .....	72
4-3. Instructor rating of importance of characteristics .....	73
4-4. Sample view created for term 1 for each module .....	81
4-5. “Where am I” view created for term 1 .....	81
4-6. Participation view created for term 1.....	82



4-7. Progress view created for term 2 for entire course .....	82
4-8. Participation view created for term 2.....	83
4-9. Learner usage of Query Tool in term 1.....	84
4-10. Learner usage of Query Tool in term 2.....	85
4-11. Usage of Query Tool by groups.....	85
4-12. View usage in term 1 .....	86
4-13. View usage in term 2 .....	86
4-14. View usage timeline in term 1 .....	87
4-15. View usage timeline in term 2 .....	88
4-16. Usefulness of the Query Tool .....	88
4-17. Reasons for using Query Tool .....	89
4-18. Usability of the Query Tool.....	91
4-19. Time taken (in seconds) to generate views in term 1 .....	92
4-20. Time taken (in seconds) to generate views in term 2 .....	93
4-21. Accuracy of the Query Tool .....	94
4-22. Learner perception of Query Tool effect on learning .....	96
4-23. Query Tool usage and postings made in iHelp Discussions in term 2 .....	98
4-24. Query Tool usage and postings read in iHelp Discussions in term 2 .....	98
4-25. Learner motivation and perception of performance .....	99
4-26. Privacy protection provided by the Query Tool .....	101
4-27. Identifying another learner using the Query Tool .....	102
4-28. Importance of characteristics .....	103
4-29. Level of openness to other learners, friends, and instructors.....	104
5-1. Social awareness in iHelp courses .....	112
C-1. Example purpose created using Query Tool interface .....	138
C-2. Code listing of the view for the example purpose.....	139
C-3. Code listing of the model for the example purpose .....	140
F-1. Example of a table view in iHelp Courses .....	169
F-2. Edit courses menu of iHelp Courses .....	170
F-3. View creation interface.....	170
F-4. Adding characteristics to a view.....	171
F-5. Adding extra information to the view using parameters .....	172
F-6. Adding a second characteristic to the view .....	173
F-7. Adding a filter.....	175
F-8. Giving other users access to a view.....	176
F-9. Describe the purpose of the view and save.....	177
F-10. List of views for a test course.....	177
F-11. A generated view .....	178
F-12. Actions menu of iHelp Courses.....	179
F-13. Listing views using the iHelp Courses actions menu .....	179
F-14. Listing views for editing.....	180
F-15. Editing a view .....	181
F-16. Setting privacy .....	182

# 1. INTRODUCTION

The use of learner models (representations of learners as they use a system) in intelligent tutoring systems (ITSs) allows the ITS to personalize the interaction with the learner. In an e-learning environment where learning takes place using the Internet, personalized content increases learning speed and effectiveness by providing learners with material tailored specifically to their needs. The goal of work surrounding personalization in e-learning is to provide what the learner needs to see, when he or she needs to see it, and how he or she needs to see it [BRUSILOVSKY1999]. The learner model is important to this process, as characteristics in the model can provide insight into this personalization process.

Most learner models are used by the ITS and are not shown to the learner. However, the learner modelling community ([BULL1999a], [KAY1997], and [DIMITROVA1999a]) has looked at opening the model (called an open learner model, or OLM) to the students to whom the model belongs, their classmates, or their instructors. In an open learner modelling system, the learner is brought into the modelling process, either by viewing the contents of his or her model, or by interacting with it. By giving the learner control over the model as it is developed and used by the ITS, the learner gains control over his or her learning, providing a greater personalized interaction within the e-learning environment.

One approach to generating learner models is active learner modelling. In an active approach, a complete model is not created or stored in one place for all system methods to interpret as required. Instead, a context-specific model is created as it is needed, for the particular goals of the system at the time. This provides the benefit of having an up-to-date model without needing to use complicated updating

algorithms on a model which may not even directly suit the system's needs [MCCALLA2000] [VASSILEVA2003].

When looking at these two learner modelling philosophies and how they can be used together, a question arises: what does it mean to open a model if there is no model (as in active modelling)? Initial analysis of this question suggests that the purpose and context (particular social and cognitive factors) play a large role in the development of the learner model and common purposes can in fact be developed so the model can be opened. The learners want particular information from their models – if they are able to directly query the learner model the ITS can realize their purposes and produce the models they are after.

The main research question of this thesis is to determine if a method can be developed to compute an open model delivered actively to the learner to suit the learner's needs. Specific requirements identified are:

- To build a Query Tool to be used by learners and instructors to explore their learner models based on questions the learners have about social and cognitive aspects of the models.
- To use active modelling to compute the learner models and develop appropriate visualizations.
- To maintain a balance between system usefulness and privacy protection as the learner model is opened to instructors and peers as well as the learner for comparative purposes.
- To test the Query Tool and learner models in terms of performance, robustness, and appropriateness in real-world learning environments.

Contributions of this work to the open modelling community will be to identify additional mechanisms to open the learner model to the user using additional context to target the learner model as required to the user's needs in viewing it. Additionally, exploration of privacy and the impacts on an effective open learner

model will be another contribution of this work to the community. Contributions to the active community will largely be an addition of a different area (open modelling) in which to examine context and purpose in an open environment.

In Chapter 2, I discuss in more detail the related research in open modelling and active modelling as each philosophy applies to e-learning environments, and present some initial discussions when combining the two into a method referred to as “Active Open Modelling”. In Chapter 3, I introduce the Query Tool developed to explore this method and present examples from real courses to show the capabilities of the system. Chapter 4 outlines a pilot study completed to analyze different aspects of the Query Tool in online and blended learning courses. Chapter 5 then summarizes the capabilities of the system and the survey findings, and describes possible future directions following the conclusion of this research.

## 2. LITERATURE SURVEY

This chapter describes current and past research in intelligent tutoring systems and user modelling with a focus on the open and active modelling approaches. The goal is to introduce the research questions that underlie *active open learner modelling*.

### 2.1 *User Modelling and Learner Modelling*

User modelling is the method of tracking information about specific users of a system and using the information to determine certain characteristics of the learner. The user model is then used to adapt a system to the user in order to provide an individualized interaction with the application. For example, user modelling may be used to adjust display settings on webpages [KOBSA2001] or provide individualized instruction to learners in an educational environment (also known as an Intelligent Tutoring System, or ITS).

Learner modelling, a subset of user modelling where learners are modelled in an ITS, has its own specific research questions. The process involves recording information while a student interacts with the educational system and inferring characteristics (or beliefs) about the learner, such as his or her knowledge level in a particular area. A learner model contains variables important to the domain or to the system in order to modify the environment to meet the system's pedagogical goals. There can be few or many variables in the model, including tendencies in behaviour, strategies and goals the learner currently possesses, and the learner's understanding of different topics. What is in the model relies heavily on the domain and the goals of the system. Therefore, variables may have numerical values, be assigned in categories, and change over time as the learner interacts with the system

[MISLEVY1995]. The reasons a system needs a learner model are equally varied, from simply understanding or predicting a situation in an educational environment, to adapting aspects of an intelligent tutoring system to match characteristics observed in the learner model, to using the learner model in creating more suitable learning materials [BAKER2000].

Two specific approaches to learner modelling are examined in this chapter: open learner modelling and active learner modelling.

## *2.2 Open Learner Modelling*

While most systems keep the learner model hidden from the user, recent research has explored the benefits of opening the model up to the users of the system, so-called open learner modelling [MORALES1999a]. In an open system, the learner is brought into the modelling process, either by viewing the contents of his or her model (“scrutiny” as in [HOLDEN1999]), or by interacting with it [DIMITROVA1999a]. These two aspects have started to merge and the term “open learner modelling” now applies to both.

### *2.2.1. Viewing or Scrutiny*

In an open system, the learner can view the contents of the model, meaning he/she can see what characteristics are captured in the learner model (and compare them to his/her own perceived values of these characteristics [KAY1997]). In an open learner model, learners want to see such information as concepts they understand, problem areas on which they should focus, and possible misconceptions [BULL2007b]. By making the characteristics available to the learner, he/she can determine what characteristics are important to the system, and determine why the system has behaved the way it has (providing extra help when a misconception is identified, etc.). The learner model is at the core of an adaptive educational system; giving the learner control over the learner model by making the model and modelling process transparent gives the learner more control in the environment, and thus control over his or her learning [KAY2001]. Scrutiny of the model also enables

the user to see and “appreciate” what personal information has been captured by the system and determine what else has been deduced from it [KAY2006].

Opening the learner model to scrutiny introduces several questions about the learner model and the underlying data: how should the system interpret the evidence available in a system and how should the system combine multiple evidence sources to come to higher level conclusions about the learner [KAY2005]? In an online course delivery system, page views may provide evidence the learner has read through the material, and answers to multiple choice questions test the learner’s understanding of the topic. However, is this enough to say with certainty that the learner knows the topic? Even once the system determines the learner knows the topic, is the internal learner model of the system appropriate to open to the learner as it is? [BULL2007]

A related, but more complex idea is opening up the learner model to the scrutiny of others: peers, tutors, and teachers. This requires a different approach, as users of a public learner model have different purposes than a user viewing his or her model. For example, a learner may want to compare his or her knowledge levels against those of his/her peers (an expert or an average student), as in Bull and Nghiem [BULL2002b] or against an entire class through a specialized group model during collaboration [TONGCHAI2005]. Or a learner may want to view a potential helper’s characteristics in order to choose the best helper from a list [BULL2001]. Such a system would even allow a helper to tailor the response when giving help after a request, as in Collins et al [COLLINS1997]. Of concern here is to provide useful information for the viewer’s needs, designed specifically to what he or she wants to discover from the model, or to provide the opportunity for the user to find and choose the characteristics he or she would find to be the most helpful. Studies have found that learners have indicated that viewing others’ learner models has been highly useful to them [BULL2005] [BULL2005b].

Opening the learner model to instructors is a special case to consider, particularly as the open community has tended to focus on models opened to learners instead of instructors. Instructors typically require more information about learners in order to judge progress and assign marks. Instructors need to know what the learner's problems and needs are in the course as related to individual students, groups of students, and the entire class [KOSBA2005]. Instructors may be interested in knowing the effectiveness of online course materials which an open model also provides [JOVANOVIC2006].

Opening learner models to peers and instructors also opens issues related to privacy, as learners may be sensitive to information included in the model and may not wish to share it to others. Initial studies by Bull et al [BULL2005] have shown that student acceptance of opening learner models to peers and instructors varies between learners. These studies asked learners to specify whether their entire model could be opened, either anonymously or publicly, to different people, such as other peers or instructors. The majority of learners opened the learner models to peers and instructors, and only about half of those were anonymous. Overall, the students appreciated the opportunity to specify exactly who may see their entire learner model, but the study did not explore reasons or patterns of why the learners opened their models to others in their community (or why they did not).

### 2.2.2. Interaction

Open modelling need not consist solely of viewing the model, but may also involve interaction with the model, depending on the goals of the system in opening the learner model. In this approach, the student can not only view the information contained in the model, but can influence or change parts of the model or modelling process [BULL1997]. This allows the student to determine what the system has discovered about him or her and challenge parts of the model if seen as inaccurate. Instructors and even parents may also add more information to the learner model in a process called "active reporting" by Zapata-Rivera et al [ZAPATA2005].



The interactive process of open modelling is often described as an equal dialogue between the user and the system: each is given the opportunity to express beliefs about the learner's progress and provide evidence as the learner model is developed ([DIMITROVA2002] and [DIMITROVA2003]). Because the learner is able to view and manipulate the information stored within the model, the learner has even more opportunity to reflect on the learning process than when merely viewing the model. Moreover, perhaps the learner is also more motivated to achieve the learning goals supported by the system [MORALES1999b], [DIMITROVA1999a], and [DIMITROVA1999b]. Research has shown that learners are able to accurately assess their own domain competence [BRNA1999]. This suggests that they should be given the opportunity to express their opinion within the system. However, recent studies from the Adaptive Hypermedia community have shown that when users try to fix perceived incorrect user profiles automatically generated by the system, there is only a little influence on the accuracy of the models, as users felt the profiles still did not represent them [WAERN2004]. Further, user input into the models appears to harm system effectiveness in some cases [AHN2007] and lead to inaccurate learner models in other situations where learners are allowed to create their own models from scratch [MABBOTT2007]. Even so, users preferred interacting with these open models, and in general the community agrees interaction and user input into the models is important, though the effects should be studied further.

### 2.2.3. Motivation for Open Modelling

There are many reasons to open the learner model to the users of a system, including: encouraging reflection, validating the model contents, and assessing the learner's progress [VASSILEVA2003].

#### 2.2.3.1. Reflection

The primary reason to open the learner model is to encourage reflection [BULL1997]. Reflection is a meta-cognitive process that means to "think about" one's progress and goals. As a student views and/or interacts with the information

within an open learner model, he or she reflects on his or her model's characteristics as described by the system, thus gaining a greater understanding of the domain and his or her current beliefs and performance within the system [BULL2002b]. "Involving learners where they can inspect and discuss their models is a reflective activity which leads learners to articulate, validate, and challenge the robustness of their own domain competence" [DIMITROVA2001]. Studies in systems with open models have shown that an open model encouraging reflection may have improved the performance of less skilled learners and increased the confidence of more skilled learners [MITROVIC2002], particularly when the reflection is guided through the aid of an instructor or even an artificial agent [ZAPATA2003].

#### 2.2.3.2. Validation

Previous research has shown that solely computer-based observation of a user results in a less accurate model of that user [DIMITROVA1999a]. Instead, a two-step method of system observation and user interaction provides a balanced, representative view. Opening the model to the learner helps to answer questions such as "what does the system know about me", "how did it come to these conclusions", "what is the meaning of the parts of the model", and "how can I control my model" [HOLDEN1999]. The learner is often able to accurately assess himself or herself and can accurately defend why he or she thinks a different rating or analysis is deserved [BRNA1999]. By bringing the learner's rating of themselves into the system, the diagnosis becomes more accurate as more evidence is provided to reflect the "true situation" [BULL1995b].

Other users in the system can also provide ways of validating the learner's model. For example, in the I-Help system, a measure of helpfulness is computed based on the user's own analysis and other peers with whom the user has interacted [BULL2001]. In this case, the other users have a more accurate impression of the learner's true helpfulness. When helpees and helpers mutually inspect each other's knowledge profiles in PHelpS, inaccuracies in the models can be clarified by each learner's analysis of the profile [COLLINS1997].

#### 2.2.3.3. Assessment

Opening the learner model to teachers or teaching assistants has also been explored in order to aid assessment or to adjust teaching methods [BULL2002b] [VASSILEVA2003]. In this case, an evaluation of the student is done, either to assign a grade based on the learner's performance, or to determine how well (or poorly) the learner is doing within the system, which can diagnose particular problems or misconceptions. Viewing more than one model allows the instructor to identify common misconceptions and alter teaching methods to suit the needs of the students in the class.

#### 2.2.3.4. Other Motivations

Other motivations exist to open a learner model as well. For example, an open learner model may allow for collaboration and competition as the models are opened to peers in a community [BULL2007]. Or, the learner's right to information about them and control over the information can be enabled by opening the learner model [KOBSA2002]. Students and instructors have indicated that they appreciate having access to the learner models as the models increase understanding of the domain and increase meta-cognitive skills [MITROVIC2007]. There are many reasons to open a learner model that depend on the situation and the pedagogical goals of the system.

#### 2.2.4. Visualization of the Open Model

How to visualize the model information must also be considered when opening up a learner model. The literature shows there is a wide set of visualizations used to represent learner models, though overall there appears to be no standard way to open learner models [BULL2005]. The most common visualization is a graphical skill meter which shows a learner's knowledge level filled in as a percentage bar of the expected knowledge level, either with respect to an expert [WEBER2001] or another measure such as how the learner compares to others in a learning community [LINTON2000]. Other formats have also been explored, including textual

representations [BULL1995b], tree structures of concepts [KAY1997], concept graphs [DIMITROVA2003b], and graphical representations of Bayesian networks [ZAPATA2000 and ZAPATA2004]. More recently, exotic open models have also been explored as better ways to visualize and interact with learner models. Virtual pets have been explored as a way to encourage children to monitor the health of their pet, where the pet's health is a visualization of the child's progress in learning language [CHEN2007]. Haptic models [BULL2005] use 3D representations to indicate qualities of touch to represent characteristics of the model (for example, a soft item is not as well known as a hard item). While a study of the work found that only a minority of students preferred the haptic models to more traditional models, work like this shows the diversity of visualizations in the open modelling area.

Three main criteria for an effective open model have been identified: understandability, ability for effective inspection, and reduction in the cognitive load [RUEDA2003]. To meet the needs of the user and to encourage reflection and interaction with the model, the information must be presented in a clear, unambiguous way. Overall the context and situation dictate the best method [ZAPATA2000]. The granularity at which the characteristics are modelled or displayed is also important—having a great amount of detail may be confusing to the learner, or may not be needed for a particular use [MORALES1999a]. As well, perhaps in some situations, it is better from the student's perspective to not reveal parts of the model, as this may discourage reflection or learning [KOBASA1990]. Or the model may be too complex to share all details effectively with the learner [MORALES1999b].

#### 2.2.5. Summary of Open Learner Modelling

The research in the open learner modelling community covers many different areas and techniques in considering reflection, motivation, and visualizations of the open model. Authors in the community have commented that there is no standard way to open a learner model or analyze an open model, partially due to the complexity of issues surrounding opening the learner model to learners and others [BULL2007].

### *2.3. Active Learner Modelling*

Open learner modelling includes the learner in the production of the learner model. Another modelling approach, active learner modelling, also includes learners in the modelling process, but takes a different perspective on what it means to model a user.

Most user modelling systems maintain a data structure which stores the information in the model but does not decide how the information is to be used. For example, a stored model may contain calculated fields such as “knowledge level” or “helpfulness”, which are updated periodically as a student interacts with the system. The stored number is a compilation of all previous interactions with the student. In such systems, the application decides how to interpret the results of the model when it is needed.

Even Fink and Kobsa [FINK2000], who have a distributed model across different systems and servers, consider the model to be virtually centralized, similar to a distributed database. In contrast, active modelling in the original agents-based I-Help system had a virtually distributed model which is in reality stored in a centralized database [VASSILEVA2003] [VASSILEVA1999]. Instead of interpreting the results of a pre-computed model, an exact model for a particular purpose is created when it is needed, allowing for more precise modelling for a certain context. Thus, the user model can be adapted to the requirements of the current user and purpose, using fragmented information from a variety of sources [MCCALLA2000].

#### *2.3.1. A Definition of Active Modelling*

Active modelling views user modelling as a process, where the model is computed for just-in-time delivery when a particular need arises. There is not one large model, but many fragmented models and/or pieces of raw data that are retrieved, integrated,

and interpreted according to the task and context. So, the active model can be thought of as a function:

$$\mathbf{learnerModel: f(s, o, p, r=\{r1,r2\})} \quad (2.1)$$

with the following parameters, the context for the modelling:

- *s (subject)* = the agent (software or human) doing the modelling
  - *o (object)* = the humans (or agents) being modelled
  - *p (purpose)* = the purpose for which the model is being created
  - *r (resources)* = the sources of information or constraints for computation
    - *r1 (referees)* = other agents contributing to the modelling process
    - *r2 (resources)* = computational resources (time, computing resources)
- [VASSILEVA2003]

In the active approach, the focus is on developing clichés that capture typical learner modelling computations. Research into the nature of such clichés suggests that they are often oriented around the various purposes (*p*) that underlie the learner modelling computations [NIU2003] (so-called purpose clichés).

The concern in active modelling is trying to make sense of too much information [MCCALLA2000]. In traditional systems, only limited information is used to interpret or diagnose misconceptions or characteristics of the learner. An active model can potentially contain vast sources of evidence to make the calculation “perfect”. However, real-world constraints such as time and computation restrictions force a non-perfect result to be found instead. For example, any-time algorithms [NIU2003] provide a method of calculation for varying lengths of time where the more time available to the algorithm, the more accurate the result, but a result is always returned. In learner modelling, where diagnosis is key, this added

accuracy is not always required, and the effort not justified [SELF1994]. So in active modelling, it is important to find adequate sources of information to find the result, and eliminate information which does not directly apply to the computation of the model.

#### *2.4. Exploring Open Modelling in an Active Context*

Summarizing the previous section's main points about open modelling and active modelling:

- The open model must be comprehensible in order for the learner to use it.
- The cognitive load on the learner must be reduced; the open model should not increase the cognitive load already on the learner when using the educational system.
- An appropriate visualization must be developed for the context.
- Learner models are often large and make use of various methods to update the model.
- Learner models gain information from potentially many sources (also called referees in active terminology) including the system, the learner, peers, instructors, teaching assistants, etc.
- Active modelling provides a way to interpret many sources of information in context to provide a smaller, targeted model.

Both active modelling and open modelling focus on the learner. These two methods, when combined, can bring the advantages from both to produce an effective interaction between the learner and the system. The natural question is: how do you open the model when there is no model? Considering this is the active approach, this should be rephrased as: when the model needs to be opened, how do you compute it? The answer: users and purposes provide context constraints which

determine what information and interaction style may be appropriate for the user. This is explained in the remainder of this section.

One of the definitions of open modelling is “including the learner in the modelling process”. Active modelling adds to this concept (and leads to *active open modelling*) because the very modelling process can be updated for the learner in a way that the traditional methods can not do. Different algorithms can potentially be applied for different types of people. Model contents can change based on roles and other relevant characteristics of the learners. Even the level of detail (e.g. granularity [MCCALLA1994]) can change, based on a combination of characteristics of the learner and the purpose. Thus active open learner modelling is essentially a form of personalized modelling: including information in the model which is targeted specifically to the one viewing the model (the viewer) and adjusting the model specifically for them.

The viewer has specific requirements for viewing or interacting with the learner model. Knowing these requirements by analyzing the purpose and other factors allows the generation of only the information that is really required by the user (a key benefit of active modelling). It also cuts down on the amount of information the learner must process about the model by removing information which does not enhance the current purpose.

Recall from the previous section that the modelling process is a function of the agent performing the modelling, other learners, a purpose, and the available resources (Equation 2.1). Two of these factors in particular, purposes and learners, affect the context and content of the model.

#### 2.4.1. Purposes

The purpose is the driving parameter in active modelling [NIU2003]. The active model must be calculated as the purpose requires, and the purpose ultimately determines what to open up to the user and how. However, the user may have many



different purposes for interacting with the information in the model. For example, a learner may be reflecting on his or her standing in a course or alternatively trying to choose the best helper from a presented list. Even within these broad categories, there may be further refined purposes, such as comparing results in a quiz to an expert or comparing results in a quiz to a particular learner. Then, depending on the purpose, there is some information that is important and other information that is unimportant. For example, consider these scenarios:

1. The learner is comparing results in a quiz to an expert (or an average student). In this case, the learner may be interested in knowledge levels and comparisons with his or her own knowledge levels. Background information or other details may become less important.
2. The learner is comparing results to a particular learner. In this case, the background may become more important, as the student may be trying to find reasons why the other learner is performing better or worse compared to him or her.

#### 2.4.2. Learners

Another main contextual element is the users involved in the modelling. In traditional ITSs, there is just one learner, and the system adapts to that specific user. Of course, it is becoming increasingly important that a system also supports more than one learner—perhaps an entire virtual community (e.g. I-Help [GREER2001]). Thus, these other learners have an increasing effect on the learner model.

In the active approach there are three types of users important to the context: who is doing the modelling, who is being modelled, and who is supplying the information. The person doing the modelling is the most important, as he or she determines the purpose for calculating the model. So the information should be generated/displayed based on this user's characteristics which again vary according to the purpose.

Moreover, the user could be playing a number of different roles, such as a learner, a teaching assistant (TA), or perhaps an instructor. This role, if determined, helps to

discover the user's purpose and provides context to open the model. Each learner plays different roles at different times, and may be playing more than one role at once. However, for simplicity, only one role at a time is considered in this work.

The relationship that exists between the user doing the modelling and the user being modelled also provides added constraints—who I am and who you are determines what sort of information I need to see about your model or have access to. For example, there may be two instructors: one teaching the learner's current course and another instructor in the department. The instructor of the course should have access to the information about the learner related to the course, but the other instructor should not. In this case, the relationship that exists between the instructor and the learner provides constraints on what sort of information is permitted.

#### 2.4.3. Motivation

There are several advantages of the active open modelling approach: minimizing storage requirements, easing the incorporation of external data, ensuring the information is updated when required, eliminating wasted time maintaining the model, and providing multiple sources for validation.

To have the same functionality as an active system, a non-active approach may have a large cross product of all the information:  $\text{purpose} \times \text{user role} \times \text{other learners}$ . This can make the learner model grow quickly with a large number of purposes and users. However, an active system may reduce storage requirements of calculated data, as everything is found as the purpose requires. As well, the information calculated can be reduced by the constrained cases (e.g. only this type of learner can access this information). For this reason, constraining the context (and purpose) is important.

Not all purposes require all the information stored in one large learner model. In fact, some information is needed rarely or only occasionally. If this information is difficult to store and needed only once in awhile, it is better for the system to access

the information from external sources only when it is needed instead of keeping a copy stored and wasting resources to update it. However, there is a trade-off between the fast retrieval of information versus storage requirements. Accessing information in a separate system may still require additional time and resources.

In traditional systems, there may be a large amount of time spent on maintaining the learner model, particularly if the information changes often as the system is used. In a learning environment, this happens frequently, as it is expected the learner's knowledge changes even while interacting with the system. If the information must be updated frequently, it is better to use the active approach, where the information is calculated whenever the model is requested, so can always be up-to-date without additional offline calculations. In the purely active open approach, there is no wasted time in maintaining the model—you are dealing solely with the information that is relevant, and finding it as a need arises.

Active open modelling gives the opportunity to provide precise, targeted modelling. Self suggests: only diagnose what you can treat [SELF1994]. This thesis suggests only diagnose what you need in a particular context. The information can be calculated and interpreted in the exact way it is needed as opposed to calculating it once and then needing to integrate it in context in a separate calculation.

Perhaps the biggest advantage of the active approach is the ability to provide multiple data sources in a calculation (as many as are needed) and still consider the context in which the data was created. For example, Interactive Open Modelling (IOM) says that while learners should have an equal voice in diagnosis, their opinions are to be maintained separately from the system's diagnosis, since these are two very different contexts [DIMITROVA2000]. Active modelling attaches this context to the calculation. As well, some characteristics of the learner (e.g. helpfulness) can only be analyzed effectively by other users in the system. The active open approach permits the user of this data in combination with the system's own computations to produce a complete picture of the learner in context.

## *2.5. Conclusion*

The active open learner modelling approach can potentially be applied to many different areas, depending on the interactions, purposes, and learners which the area presents. The next chapter introduces the Query Tool, a learner modelling system using instructors' goals to specify purposes in a purely active open learner model. The system is used to analyze the benefits and possible downsides of this approach.

### 3. ACTIVE OPEN QUERY TOOL

For online and blended learning courses, it is difficult to measure how a student is performing, and whether external actions must be taken to encourage behavioural changes to improve performance. Online delivery systems capture a large amount of raw data as learners interact with the system, from implicit data measuring the time a learner spent viewing a page, to explicit data capturing quiz answers. This data alone is too large to make sense of in any useful way. Instructors could view every single interaction with the system for a particular student, but such an activity would be tedious and time consuming. Students would not be able to use the information either – there is simply too much to go through. Thus, viewing the data alone provides no help without an established context (or meaning) behind the data.

Learners often have specific reflective questions: “what do I know?”, “what areas do I have problems with” and “what should I review” [BULL2003]. They are also curious to know how their peers are performing and how they compare: “how do I compare to the top student”, “how do I compare to an expert”, and “how do I compare to an average student” [KAY1997]. Online students as independent learners are at a particular disadvantage in knowing how they are doing when compared to in-class students, as they do not have the typical physical cues a lecture provides.

Instructors also want the ability to ask the system particular questions: “who is falling behind”, “who has yet to log on to the course”, and “is the course material effective” [JOVANOVIC2006]. One observation early in this research was that different instructors have different opinions as to what it means to “do well” in a course, making computation difficult for every possible situation. A system created

to answer these questions could thus take two approaches: establish a system-defined calculation from the system builders' analysis of many different courses that suits "most" situations, or allow instructors to define their own purposes as suited for their particular course. The second approach was chosen for this project – instructors build queries themselves using raw data and assign a descriptive purpose to the data, then run the queries to generate and view the results. Information is computed in real time when requested (in other words, computed actively), so the data is always up-to-date. Instructors may then choose to allow students to run the query as well, enabling student reflection on their progress in the course as the instructor has defined it.

To open the learner model to the users (both instructors and learners), a Query Tool was produced so the user could essentially pose questions of the course delivery system and the system would actively compute the answer to the question. In general, three contextual elements were considered in deciding how to open a learner model: the user's goal or purpose ("How am I doing", "How are my students doing"), the domain ("in Course X"), and the comparison community ("compared to other online students"). This Query Tool informally is the Compare Progress action of iHelp Courses (to be discussed below).

### *3.1. Active Open Learner Modelling Query Tool*

The active open learner modelling Query Tool had several goals at the beginning of this project. While the overall approach to the problem has evolved from the initial prototypes and examples, the goals have remained the same:

1. Allow instructors and learners to obtain a sense of where learners are in the course and how well they are doing.
2. Give the user more control of the modelling process by being involved in the modelling process.
3. Create an interactive interface to encourage the user to dig deeper into the model.

4. Provide a privacy mechanism flexible enough for the learner's privacy to be protected, yet keep a level of usability of the information presented to the viewer.

The Query Tool was built into the iHelp Courses system [BROOKS2005] – an online course delivery system used by the University of Saskatchewan's Computer Science Department to deliver blended and online courses. iHelp Courses is part of the larger iHelp suite, which includes discussion forums, asynchronous chat, and a document sharing utility for distance collaboration. Course content is linked to these communication systems, and students interact using a communication bar at the bottom of their web browser's window (see Figure 3-1). For example, the XHTML module in the course content for the CMPT 100 online course has a discussion forum only for the XHTML module, and while the learner is viewing the XHTML module, he/she sees the discussion forum for that module in the communication bar. Because of this link, interaction with the other iHelp systems can also be considered in the Query Tool and can factor into a determination of how the learner is performing, particularly if the course has a participation component, which is common in online courses. In the iHelp Courses system, the Query Tool is known as the "Compare Progress" feature and is available in the left hand actions menu of the system, as well as in the course edit pages for administrators. An open learner model is called a "view" in the iHelp Courses system.

In iHelp Courses, user access is defined by "roles". For example, an online student taking an online course CMPT 100 has the role "CMPT 100 Online Student", and is granted access to course materials based on that role. An instructor for the same course has the role "CMPT 100 Instructor". Instructor roles are assigned administrator privileges for the appropriate courses. Several student roles are possible in the same course – for example, Teaching Assistants may also be given access to the course. Thus, different groups of learners exist in the same course.

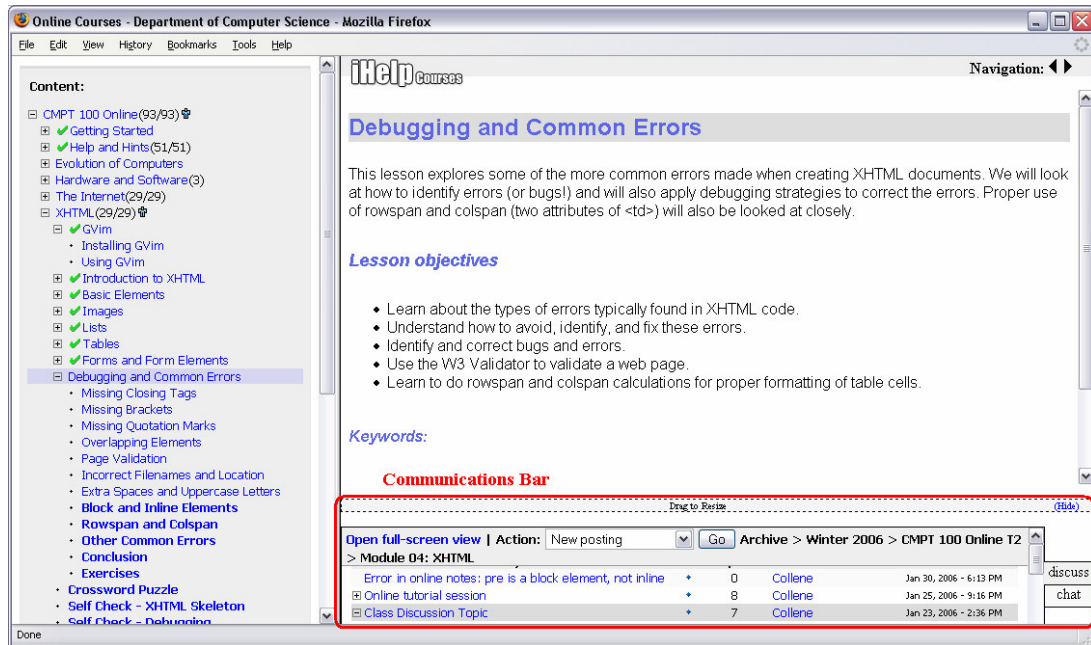


Figure 3-1: iHelp Courses and the communications bar

Development of the Query Tool started with an analysis of the existing statistics available to instructors through the iHelp Courses system to determine what characteristics and types of queries were useful to instructors. The existing tools were developed through frequent interaction with the instructor of one course. The instructor made requests to the system administrators, who then created static pages which generated overnight to produce the required data. Sample screens from this system are shown in **Figure 3-2** (with identifying information removed to protect the privacy of the users). The sample pages were used as an initial guide in the design of the Query Tool, but extended to allow instructors to specify the characteristics and learners of the tables. Further interactions with instructors of several courses and the iHelp Suite development team further expanded on this idea to produce the general system design of the Query Tool and an initial interface.



Role	NSID	First Name	Last Name	Last Log in	Last Item Accessed	Total Number Logins	Total Number Logins Past Week
all admin							
There are no people in this role.							
devel							
				Oct 11, 2005 at 09:14:45		1	0
				Feb 16, 2006 at 16:32:55	ColleneTesting2\\Basic XHTML\\Images	1	0
				Feb 22, 2006 at 10:15:59	ColleneTesting2\\Basic XHTML\\Basic Elements	2	0
				Oct 22, 2006 at 10:55:23	ColleneTesting2\\Basic XHTML\\Quiz	18	0
				This user has never logged on		0	0

Figure 3-2: Predecessor of Query Tool

### 3.1.1. Uses of the Query Tool

Instructors use the Query Tool in two different ways, which are both enabled by the system's interface discussed in the remainder of this chapter. The two uses are:

- to create queries (also called views) that only the instructor will use to monitor different characteristics about their students, and
- to create queries that both instructors and their students will be able to use for reflective purposes.


Learners use the system by viewing queries instructors have made available to them, in order to view and compare their progress with their peers.

Both of these uses have different goals and requirements, and these goals change the characteristics included in the query results. For example, a learner model opened to only instructors may contain more detailed information (such as contact information or quiz/assignment marks) that is not appropriate in a learner model opened to learners.

### 3.1.2. Creating a View

Currently, there is only one type of view that can be created – a “table view”. A table view of learners is a table that contains rows of learners and columns of learner characteristics such as names, scores on quizzes, and discussion activity. Instructors define which columns (characteristics) to include in the view by using the *view creation interface* (Figure 3-3). This interface has four parts where the instructors fill in the details of the query: the characteristics to include in the view, the learners to include in the view, the other users who can also use the view, and a description of the view. The following discussion highlights features of each of these parts of the interface by using examples from courses using the Query Tool. A full user’s guide for the view creation interface is in Appendix F.

#### 3.1.2.1. Part 1: Adding Characteristics to the View

Characteristics can be added to the table by clicking the  button on the view creation interface. For example, in Figure 3-4 the instructor has added the user id and whether the learner has viewed the course content titled “CMPT 100 Winter 2007” (the first page in the course). Notice that if a characteristic requires additional information to contextualize the query further a parameter box opens on the right hand side of the screen requesting the details. In Figure 3-4, the “viewed” characteristic (which is highlighted in yellow because it is the current characteristic being edited) requires a page to clarify what was “viewed”, so a list of course content titles appears on the right. Several other examples are shown in Figure 3-5, Figure 3-6, and Figure 3-7 for other contextualized characteristics. Essentially these characteristics are methods or processes, and the contextualizing information parameters to the methods. In fact, methods are used in the underlying code to find the requested data when it is requested. In this way, the active open modelling approach works with processes as opposed to just the data itself (as was explained in Chapter 2). Appendix A lists the complete set of characteristics and parameters currently available in the system.

**Model Table View Creation**

**Part 1: Add characteristics to the view**

**Filter Results**

Default Learner roles included:  
(If no roles are selected, the default role for the course is used instead)

- all\LCMS\CMPT 100 102 Online Helpers
- all\CMPT\CMPT Student Assistants
- all\LCMS\CMPT 100 In Class
- all\CMPT\CMPT Faculty
- all\LCMS\CMPT 100 102 Online Students
- all\Guest\CMPT Guest\CMPT 100 Online Guest

**Part 2: Specify the learners to include in the view**

**Set Privileges**

Choose who can generate this view (unchecked and excluded roles will not be able to generate this view):

- all\LCMS\CMPT 100 102 Online Helpers
- all\CMPT\CMPT Student Assistants
- all\LCMS\CMPT 100 In Class
- all\CMPT\CMPT Faculty
- all\LCMS\CMPT 100 102 Online Students
- all\Guest\CMPT Guest\CMPT 100 Online Guest

**Part 3: Specify which other users can generate the view**

**Describe Purpose and Save**

Purpose of View:

Done

Figure 3-3: Table view creation interface

**Model Table View Creation**

**Adding characteristics to a view**

**Parameters for this data type:**

Content:

- CMPT 100 Winter 2007
- Getting Started
- CMPT 100 Online Students
- CMPT 100 On Campus Students
- Browser Check
- Installing Firefox
- Enabling JavaScript
- Installing Flash
- Help and Hints
- The Course Homepage

Figure 3-4: Adding characteristics to a view

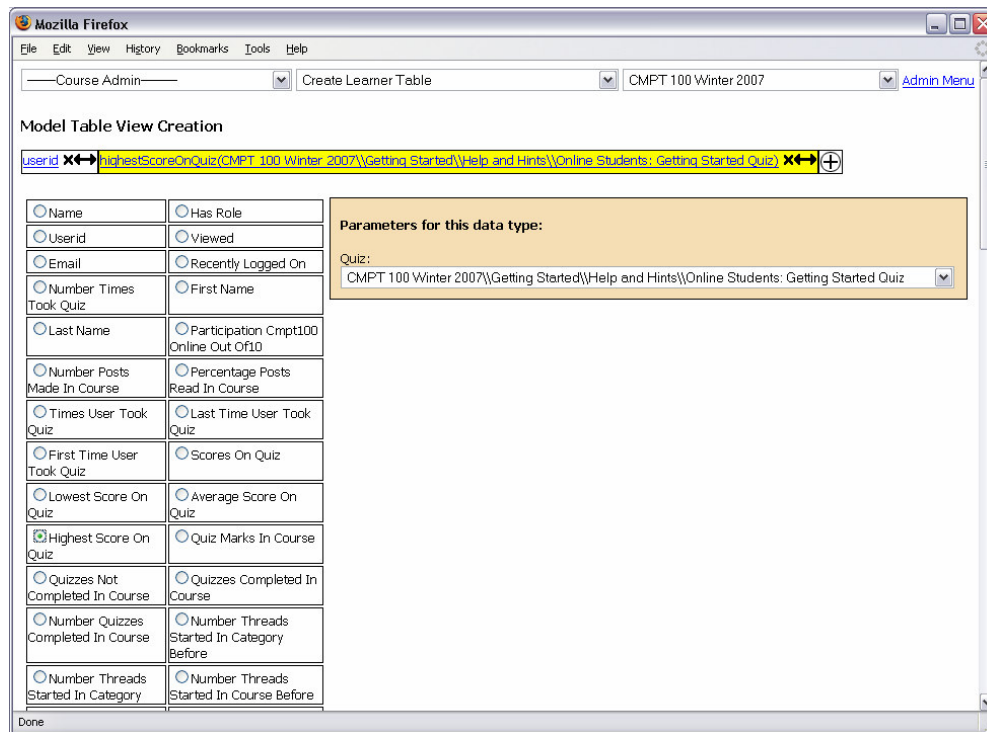


Figure 3-5: Clarifying characteristic “highest score on quiz” with a list of quizzes

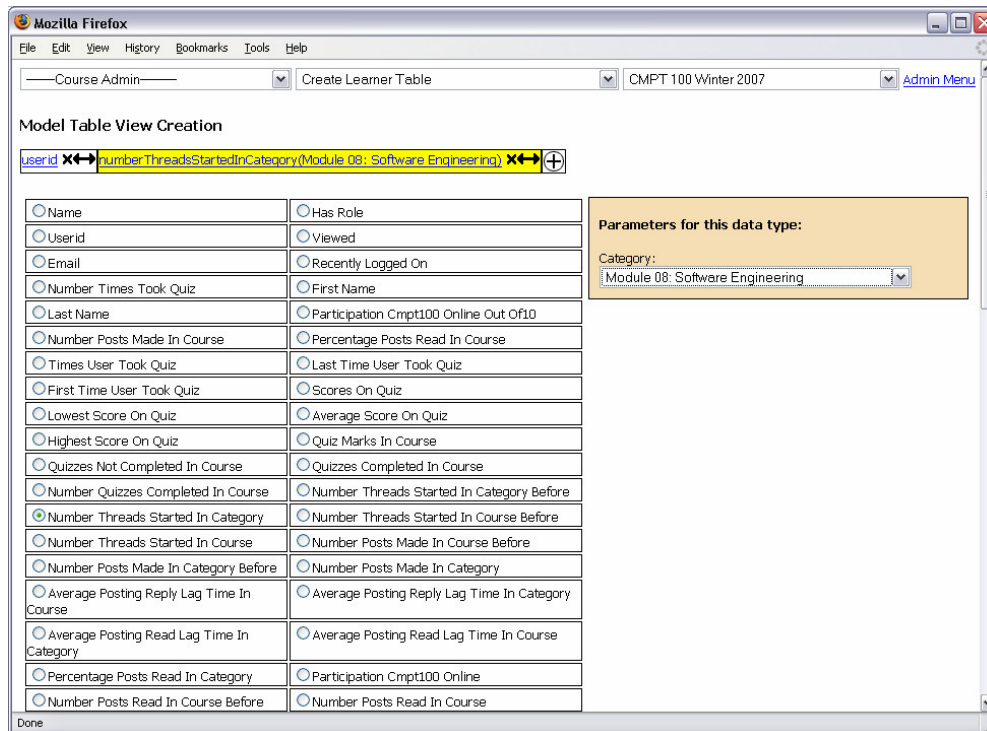


Figure 3-6: Clarifying characteristic “number of threads started in category” with list of categories

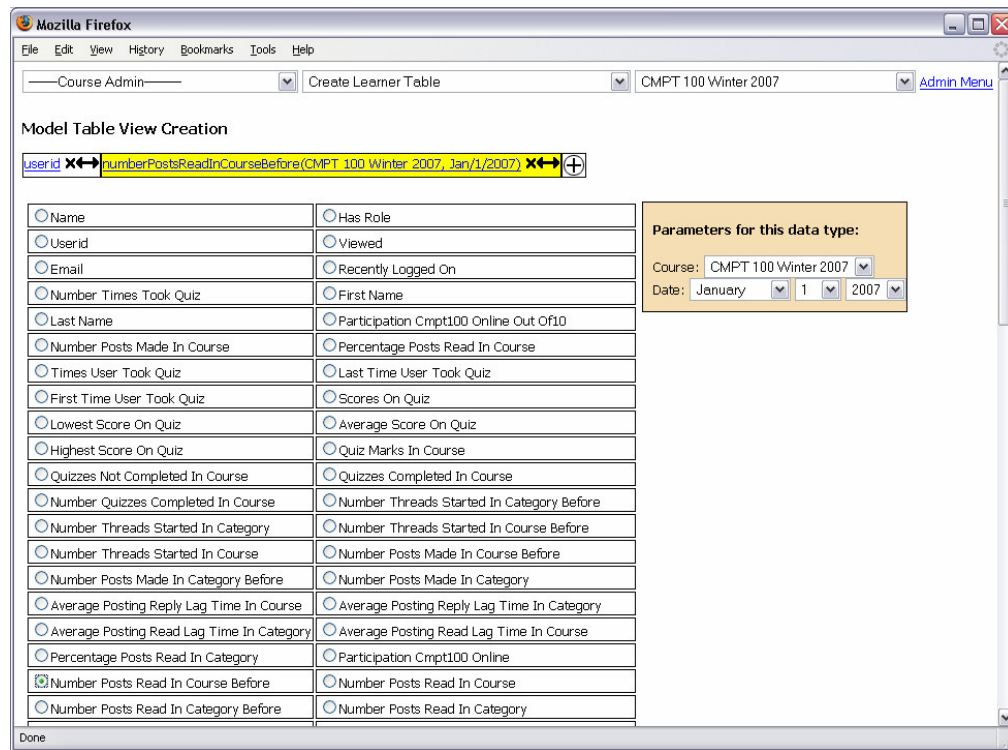



Figure 3-7: Clarifying characteristic “number of posts read in course” with list of courses and date

### 3.1.2.2. Part 2: Filtering Learners in and out of the View

Once the instructor has indicated which characteristics to include in the view by setting the columns, he/she must then indicate which learners to include in the view’s rows. First, the Default Learner roles are chosen from the list of roles which have access to the course (see Figure 3-8). This is the maximum possible set of learners who will be displayed in the final learner model when it is generated. The instructor may then limit the students returned in the view based on additional characteristics of the learners. For example, an instructor may only want to see students who have not yet logged on to a course, or those who have not completed a module. This is done by adding a filter by clicking the  button under the default learners list and choosing the filter to apply, much the same way as a characteristic is added (see Figure 3-9 for an example, and Appendix B for a full list of available filters). The filters are the most powerful feature of the Query Tool, as they provide the most targeted view of learners the instructor requires.

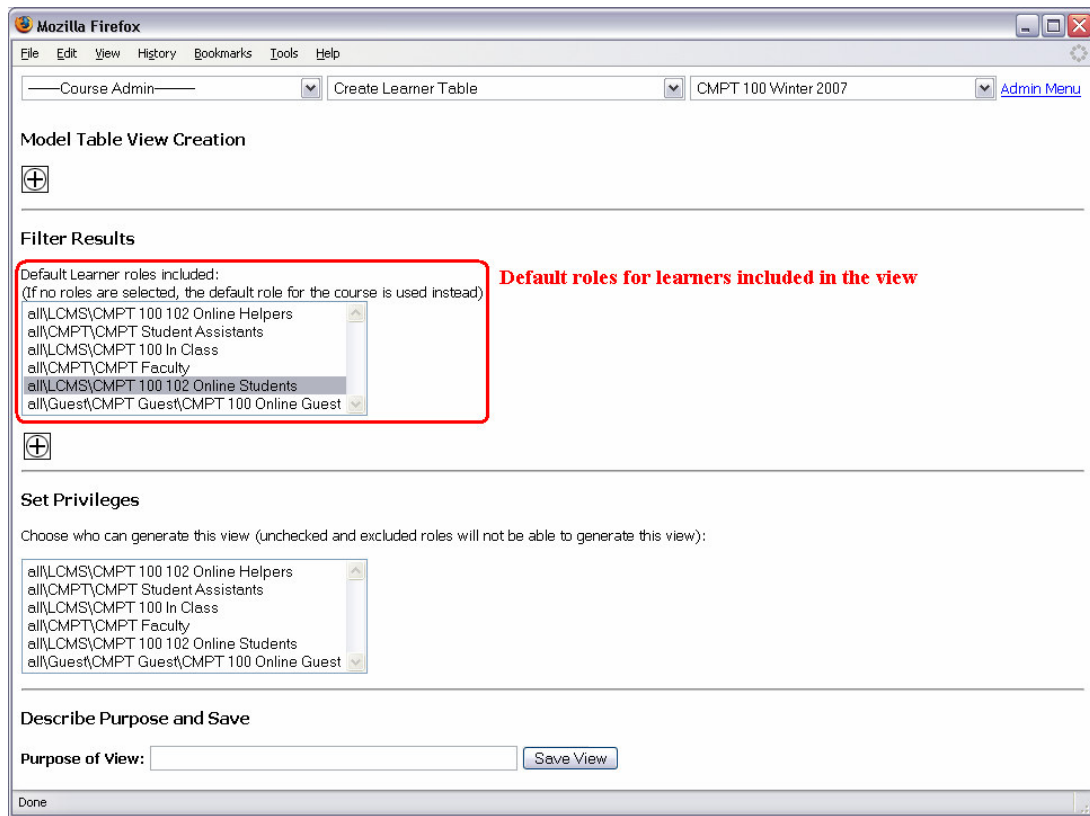


Figure 3-8: Default learners to include in the view

Filters, like characteristics, also have contextualizing parameters as needed. Several examples are shown in Figure 3-10 to Figure 3-13. First, the instructor specifies to filter students into the model, or filter students out by choosing the option from the drop down box. Filtering students in will only include learners with the filter's characteristic (in Figure 3-10, only students with the role of CMPT 100 Section 04 student will appear in the learner model). Filtering students out will exclude learners with the filter's characteristic (in Figure 3-11, only students who had *not* viewed the lecture video will appear in the learner model). Next, the instructor specifies which filter to apply, then clarifies the details of the filter. In Figure 3-10(a), the filter "hasRole" is chosen, and in Figure 3-10(b) the instructor specifies the details for the filter by choosing the role of interest, CMPT 100 Section 04, from the drop down box listing all the roles in the system. In Figure 3-11(a) the instructor adds another filter (viewed) and again specifies the details for this filter by choosing the page containing the lecture video in Figure 3-11(b). So, in this view, only learners in

section 04 who had not watched the XHTML lecture video are included in the view. Figure 3-12 shows another view with a filter limiting the view to users with a particular last name. Figure 3-12(a) shows the filter when it is initially selected, and Figure 3-12(b) shows the filter after the instructor has filled in the details. Figure 3-13 shows another view created to limit the view to include only users who have logged on since particular date.

Several filters warrant special mention here, as they provide additional functions and abilities of the Query Tool in considering the active open argument. The filter “Viewer of Page” returns only the learner who is currently viewing the page, showing the learner only their own information as calculated in the learner model. Thus, the learner model is calculated differently for different viewers of the page. The “Accessed Course Recently” filter returns only learners who have had some activity in the last ten minutes of the course. Thus, the learners returned in the model are different at different times. Context affects the results of the learner model.

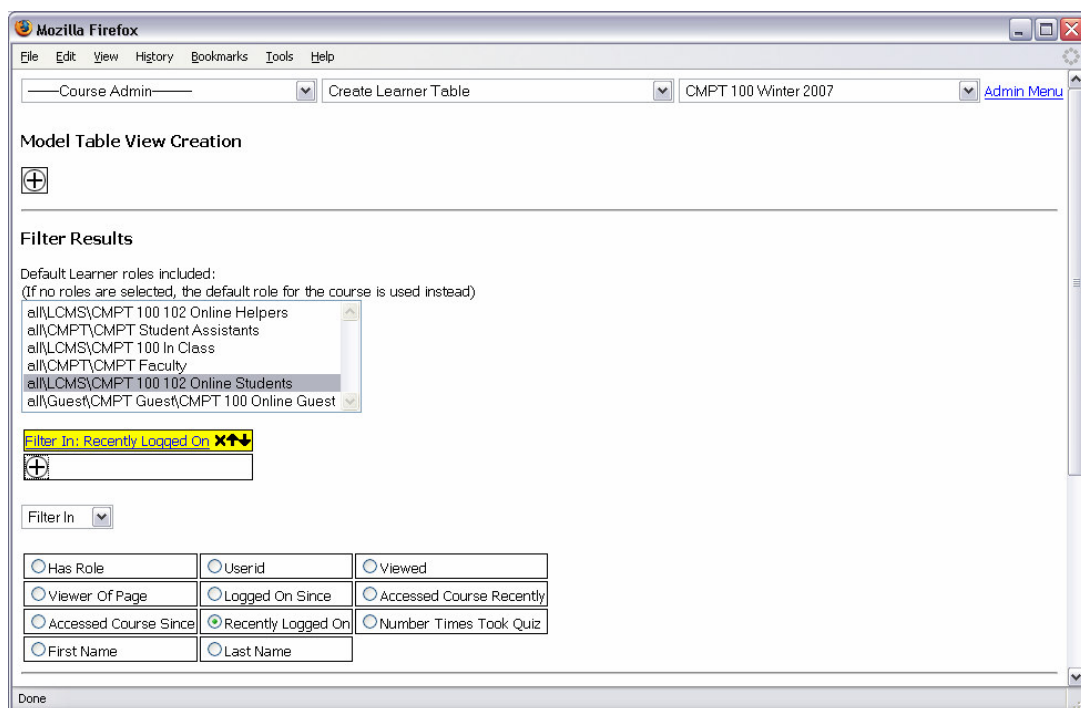


Figure 3-9: Filtering results on a view based on who has recently logged on

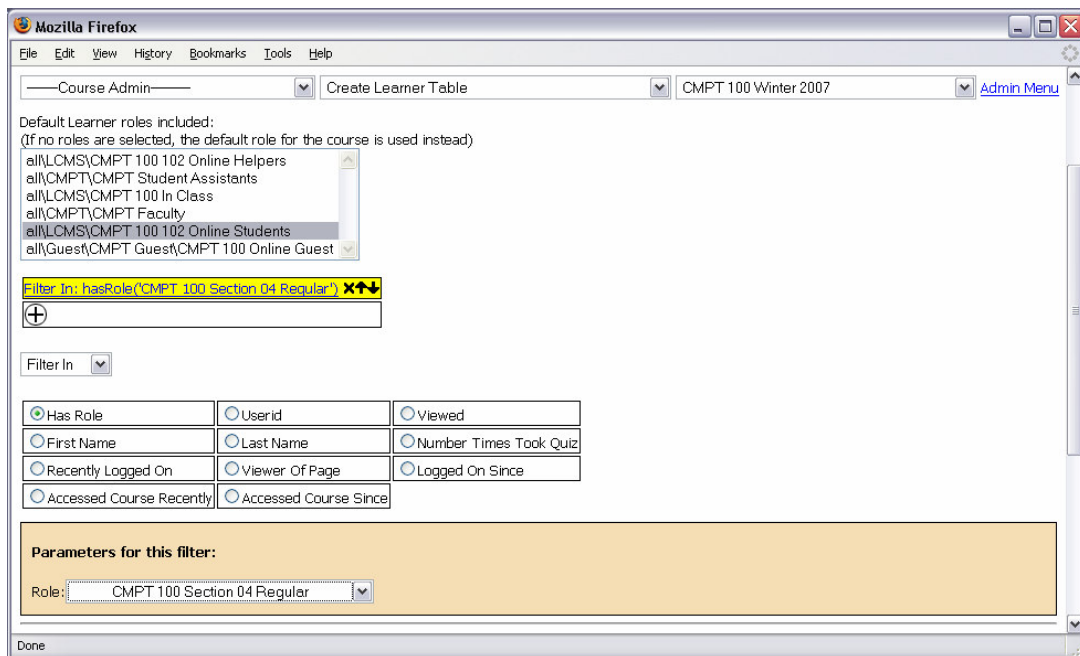
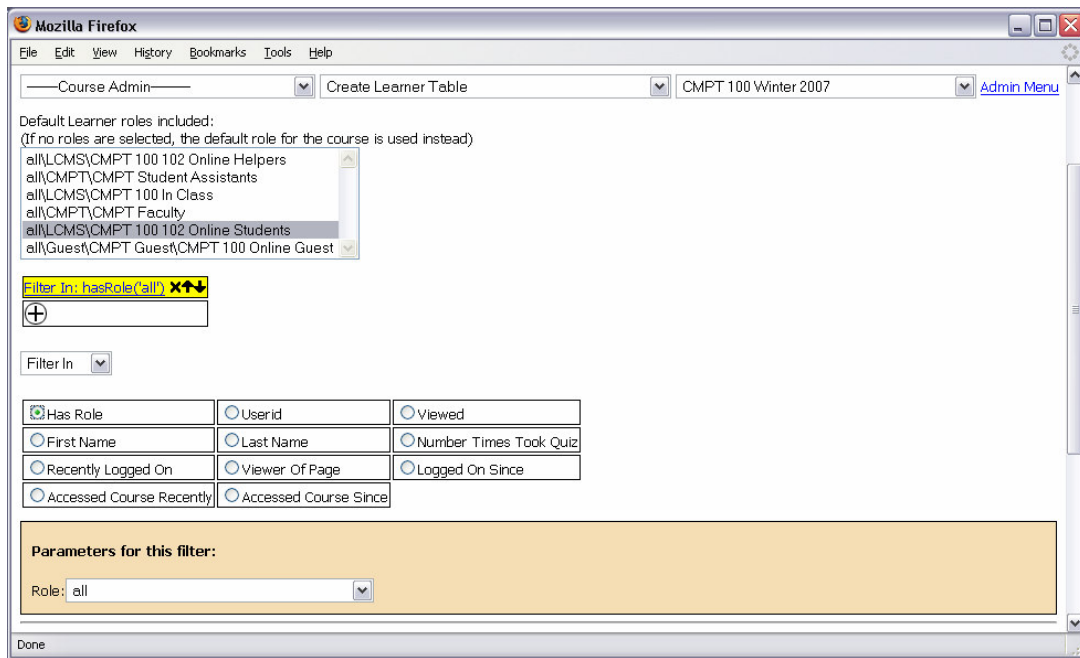


Figure 3-10: Filtering by role (a) Choosing to filter in students with a role (b) Filtering in students with the role CMPT 100 Section 04



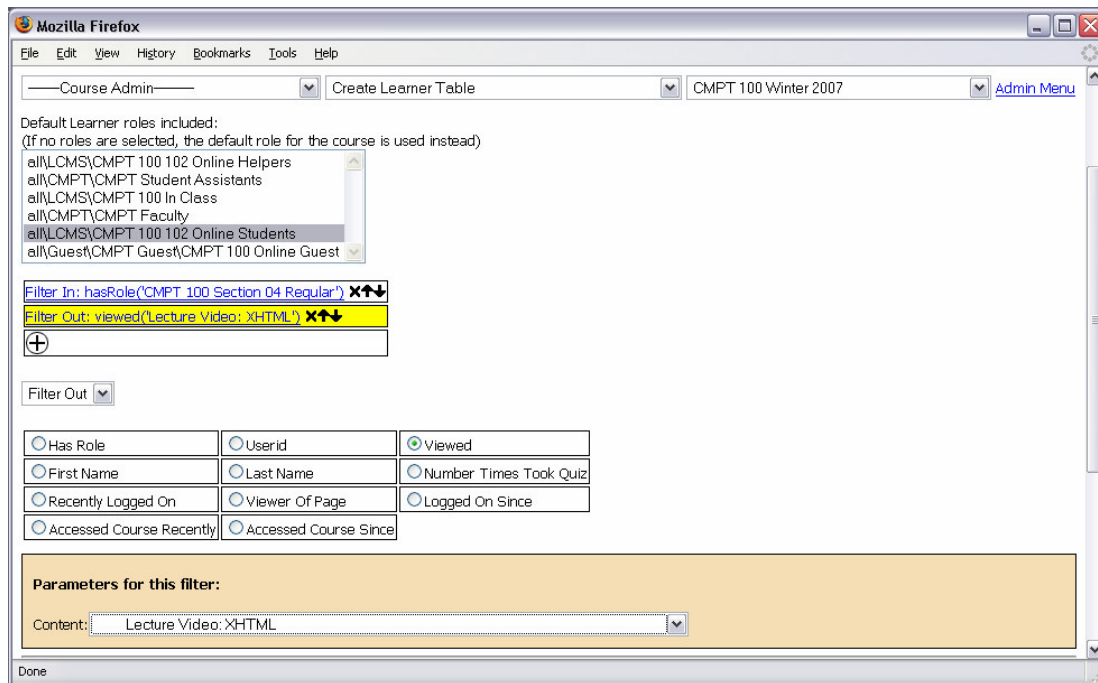
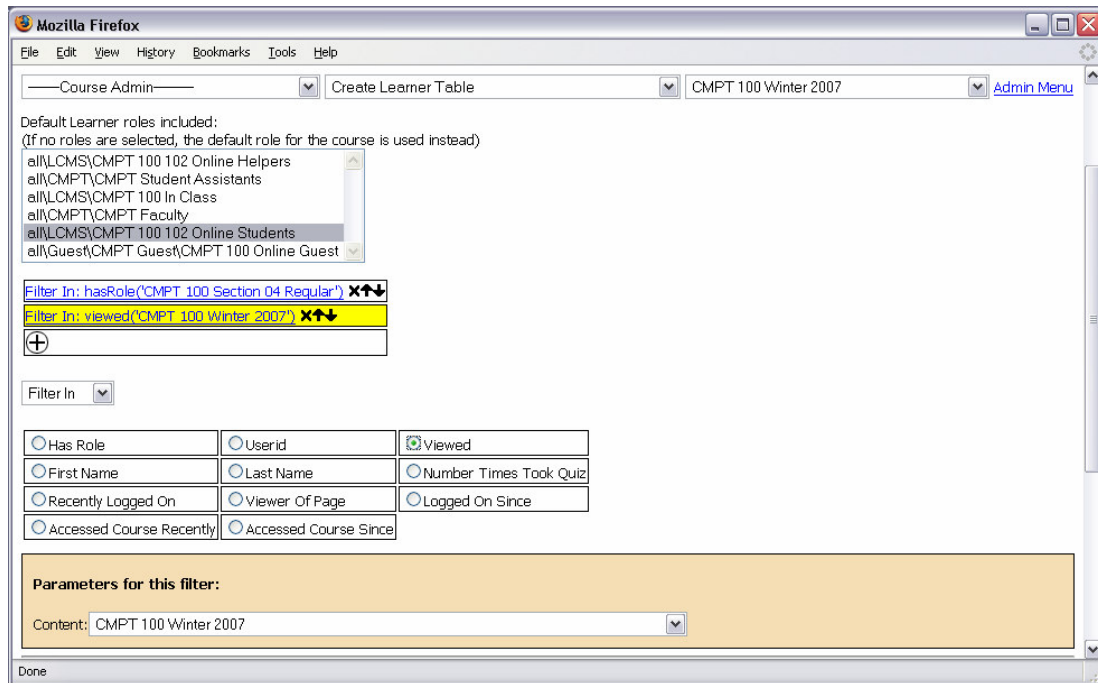


Figure 3-11: Filtering by page views(a) Choosing to filter out students who had viewed a particular page (b) Filtering out students who had viewed the XHTML lecture video

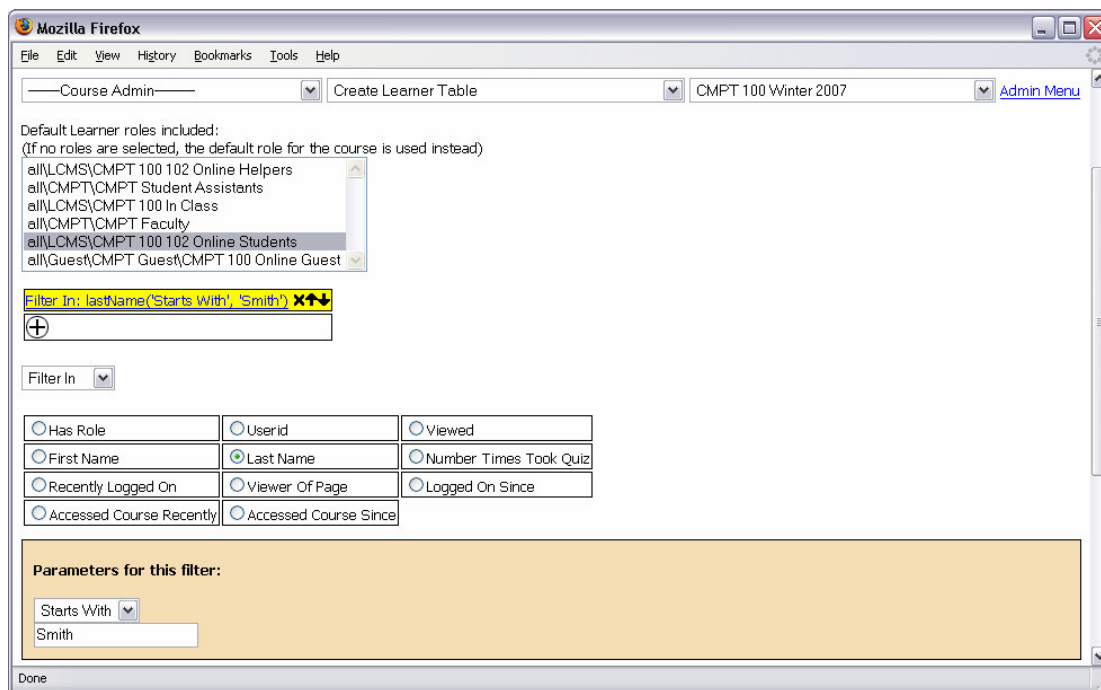
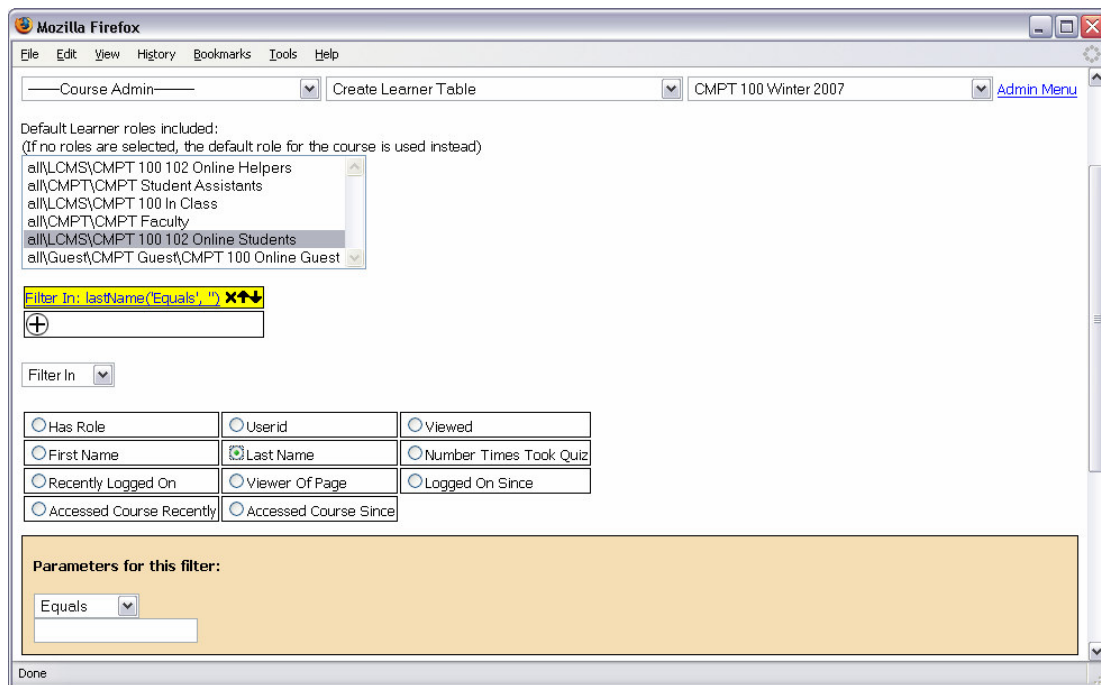


Figure 3-12: Filtering by name (a) Filtering learners based on their last name (b) Filtering in learners with last name starting with Smith

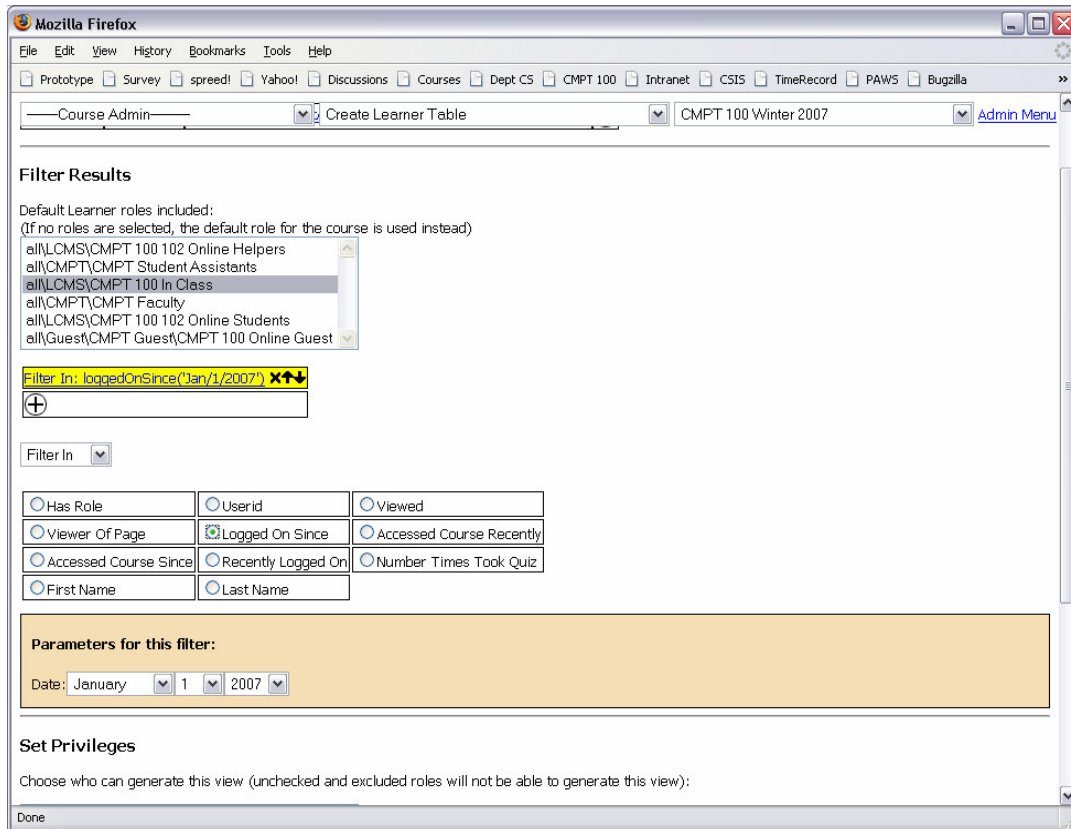


Figure 3-13: Filter in students who have logged on since January 1, 2007

### 3.1.2.3. Part 3: Opening the View to Learners by Setting Privileges

Once the view has been created, the instructor may then choose to let learners using the course generate the view by setting privileges for the view. The instructor chooses one or more roles from the set privileges select box, or leaves the options unselected if only administrators can generate the view (Figure 3-14). The view then becomes available to the appropriate learners using the Compare Progress action in iHelp Courses. Opening the learner model to learners in the course enables reflective learning as the learner is able to examine their progress compared to their peers.

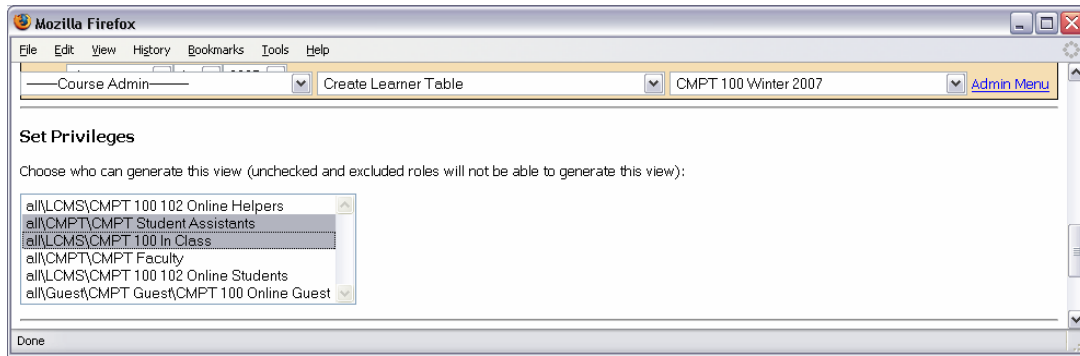


Figure 3-14: Giving access to student assistants and CMPT 100 in class students to generate the view

#### 3.1.2.4. Part 4: Describing the View's Purpose

As the Query Tool provides open-ended functionality, the instructor's specified purpose for the newly created learner model is important to describe their intention. At the bottom of the View Creation Interface screen, the instructor is asked for a purpose before saving the new view. This is the view title displayed to the users, and is often in the form of a question, depending on the audience and the purpose. Examples of descriptive purposes are: "Who has not yet logged on to the course?", "How am I doing compared to other learners in the course?", "What is the participation of the teaching assistants for Module 1?", and "Is anyone falling behind after the JavaScript module?". The instructor may choose any description they like.

#### 3.1.3. Privacy Protection

When a learner's model is opened to other peers in the course, privacy concerns arise – information generated in the model may be of a sensitive nature, and students may not want to be individually identified by their peers. However, if all information in the learner model is blocked because of privacy concerns, the learner model contains very little useful information which can be used by the viewer for effective comparisons. A balance between the two extremes must be achieved.

The Query Tool attempts this balance by allowing instructors to set privileges for individual characteristics available in the learner model for particular roles. See

Figure 3-15 for an example of setting privileges. In the figure, the roles represent the viewer of the learner model (or the subject in active modelling terms). There are two permission levels in the query tool: Allowed and Blocked. If the characteristic is blocked for a particular user based on their role, the words –BLOCKED- will appear in the column when they view the learner model. Instructors have access to all information in the model. This is another example of using the Query Tool where who the viewer is affects the results of the learner model computation (i.e. context establishes content).

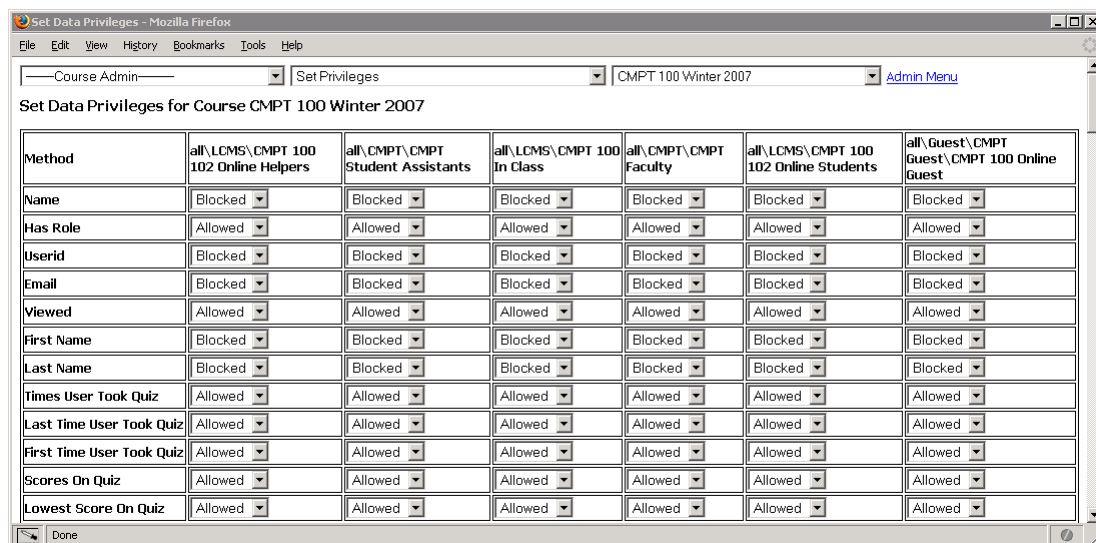


Figure 3-15: Setting privileges for characteristics based on roles

Notice in Figure 3-15 that different roles have different privileges for different characteristics. For example, CMPT Student Assistants are able to see the first and last name of the learner when viewing the generated learner models, likely to provide more effective help to the learner by allowing the assistants to know who they are, an argument made by Collins et al [COLLINS1997] to open learner models to peer helpers. However, CMPT 100 In Class students are blocked from seeing this information, as it may allow individual students to be identified by their peers.

It should be mentioned that some identifying information is blocked by default by the system if no privileges have been set by the instructor. For example, names,

userid's, and email addresses are blocked by default, and must be changed to "Allowed" explicitly by the instructor. However, most of the information in the system is displayed by default until it is changed to "Blocked" by the instructor.

The filters also have settable privileges. In this case, if the filter is blocked for a particular role, the filter is simply not applied when a user with that role generates the learner model. This is to ensure that limiting results does not limit the rows "too much" so as to inadvertently identify individual learners in the course. As in the case of the characteristics, particular filters (those relating to the userid and name) are blocked by default for all learners, and must be explicitly changed by the instructor to allowed before the filter will be applied when a view is generated.

#### 3.1.4. Listing and Generating Views

Once the query has been created and saved, the instructor may choose from the list of queries using the Compare Progress feature accessed using the actions menu of the iHelp Courses system (Figure 3-16 shows the menu and Figure 3-17 shows the list of views for the course CMPT 100 Winter 2007). Extra information is provided on the view to indicate how popular the view is, and approximately how long the view will take to generate.

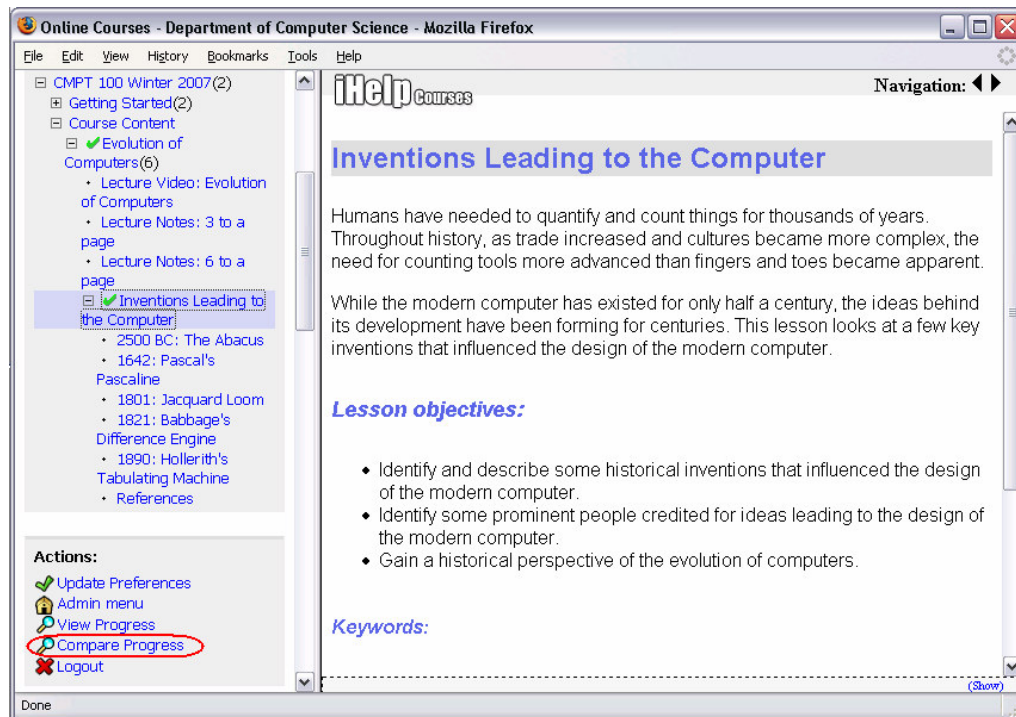


Figure 3-16: Compare progress feature in iHelp Courses

View name	View Modified Date	Times Generated	Expected Time
Do the ONLINE students prefer the lecture video or the written material for XHTML?	12 Feb 2007 02:40 PM	5	3 seconds
time spent on network videos (inclass)	19 Jan 2007 10:06 PM	5	2 seconds
ONLINE Assignment 1 tally	25 Jan 2007 09:16 AM	3	less than 1 second
Who has watched the Evolution of Computers video?	12 Jan 2007 09:25 AM	3	less than 1 second
Which students are currently online?	22 Jan 2007 03:17 PM	7	4 seconds
When should I log on to see someone online?	05 Jan 2007 02:07 PM	5	less than 1 second
who inclass saw videos module 2	19 Jan 2007 09:52 PM	1	4 seconds
Do the INCLASS students prefer the lecture video or the written material for XHTML?	12 Feb 2007 02:41 PM	2	2 seconds
Which ONLINE students are currently online looking at material?	22 Jan 2007 02:53 PM	4	2 seconds
What time of the day do learners log on?	08 Jan 2007 09:40 AM	3	5 seconds
Is there anyone in SECTION 04 who is falling behind in the material?	12 Feb 2007 02:44 PM	6	1 second
INCLASS general information - usage, login times	13 Jan 2007 11:05 AM	3	less than 1 second
Last log in times and time spent on course for ONLINE students	04 Jan 2007 08:33 AM	59	less than 1 second
What is my participation mark?	05 Jan 2007 01:54 PM	301	less than 1 second
Which INCLASS students are using the material and how much?	08 Jan 2007 09:50 AM	9	3 seconds
Contact information for ONLINE students who have not yet logged on	04 Jan 2007 08:32 AM	10	less than 1 second
How am I doing in the course compared to other ONLINE students?	05 Jan 2007 01:52 PM	176	4 seconds
viewed web and hci video	21 Jan 2007 07:42 PM	6	less than 1 second
Participation for ONLINE students	03 Apr 2007 08:47 AM	4	1 second
Contact information for ONLINE students	25 Jan 2007 09:03 AM	7	less than 1 second
time spent on hci and web videos	19 Jan 2007 10:17 PM	15	2 seconds

Figure 3-17: Created views for a course

### 3.1.5. Real Views and Generated Results for Instructors

This section describes real queries created by several instructors in four different courses. Two of the courses are completely online courses, while two of the courses are blended learning courses. Three of the courses are Computer Science courses, while one course is an informational course on university teaching practices. Each course had different reasons for creating learner models. Two of the courses used the views to establish the level of participation in the course, though each course had a different idea of what constituted participation. Several of the courses used the views to determine if students were keeping up with the online materials for the course, and offer encouragement to those who were falling behind. One of the courses used the views to determine the effectiveness of some of the course materials. Several of the more interesting queries are presented here, with generated results when appropriate. Note that any identifying information is erased from the screenshots below, although the instructor would be able to view all of the information in the views.

Figure 3-18 shows a view created by an instructor to find the contact information for online students who have not yet logged on to the course system. In this case, the instructor needed to know the email and names of the students to send them an email to log on and start the course. The view filtered out students who had already logged on. Running this query at the time this was written generated no results, as all of the students had already logged on, and thus no student was left after the filter was applied.



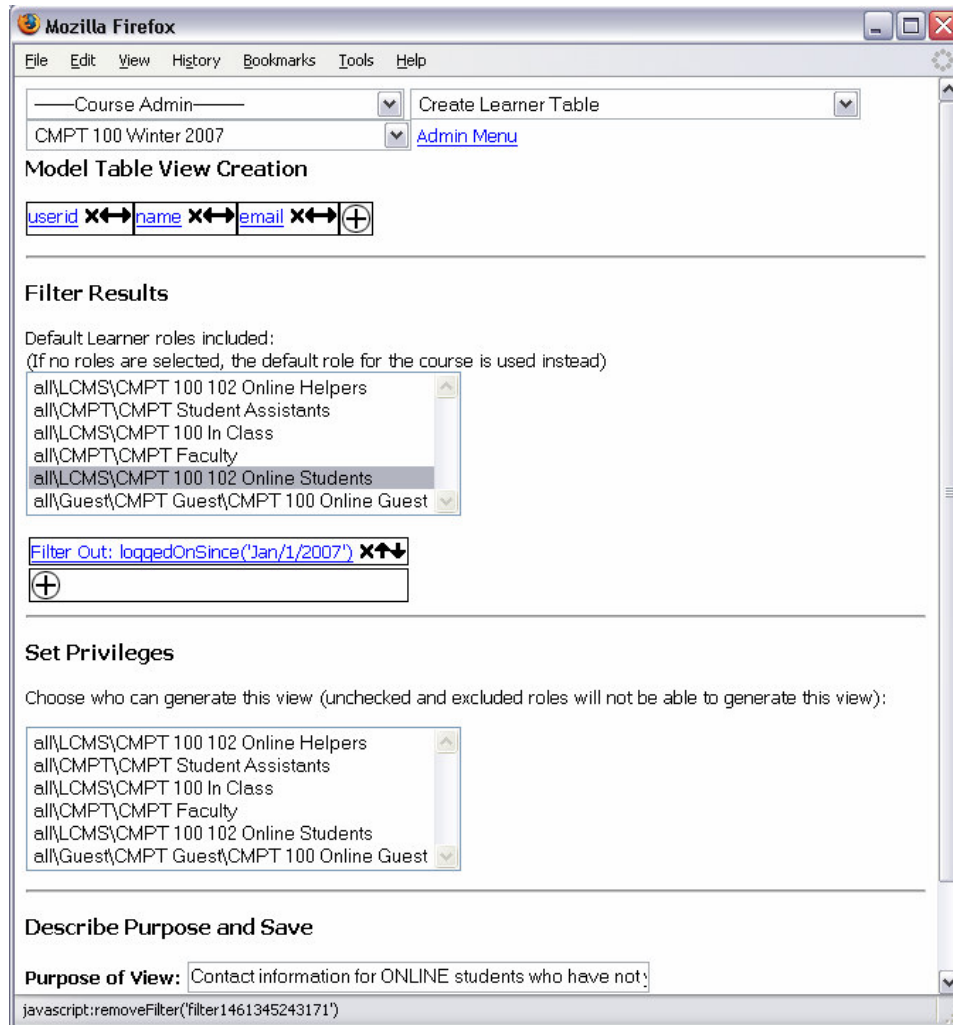
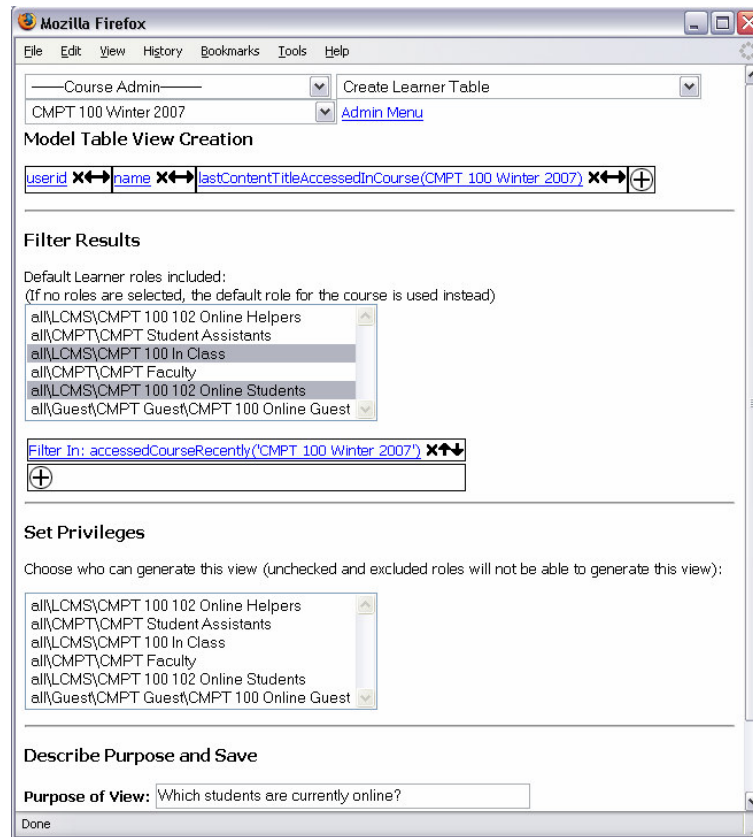


Figure 3-18: Contact information for online students who have not yet logged on

Figure 3-19(a) shows a view created by an instructor to find out if there are any students currently logged on to the system and viewing content, including which content the students were last viewing. This view filters in students who had accessed the course recently – in other words students who had viewed content in the last ten minutes. Figure 3-19(b) shows a sample of when the view is generated. In this case, four students were online. This view also gives the instructor a general idea of why the students were online and what they were looking at – in this case, one learner appeared to be catching up on lecture videos, while the other three seemed to be working on their third assignment (a programming assignment using JavaScript).



Which students are currently online?

Userid()	Name()	Last Content Title Accessed In Course(CMPT 100 Winter 2007)
		CMPT 100 Winter 2007\Course Content\Software Engineering and JS Functions and Forms\Events and Event Handlers\What are they?
		CMPT 100 Winter 2007\Course Content\Networking and the Internet\Lecture Video: Networking Part 4
		CMPT 100 Winter 2007\Course Content\Software Engineering and JS Functions and Forms\Functions\Calling Functions
		CMPT 100 Winter 2007\Assignments\CMPT 100 On Campus Assignments\CMPT 100 Assignment 3

4 learners in table

View generated in 8 seconds

Figure 3-19: (a) Which students are currently online (b) Generated results of view

Figure 3-20(a) shows a view created by an instructor to determine when they should log on to see someone online. The view shows the common log in days and times for each student in the course. Perhaps this view was used by the instructor to adapt office hours to when the students were using the system the most. Figure 3-20(b) shows a partial generated view for several random students in the course. The results shown indicate Monday is a common day students were online, but that these students didn't seem to work at a particular time of the day.

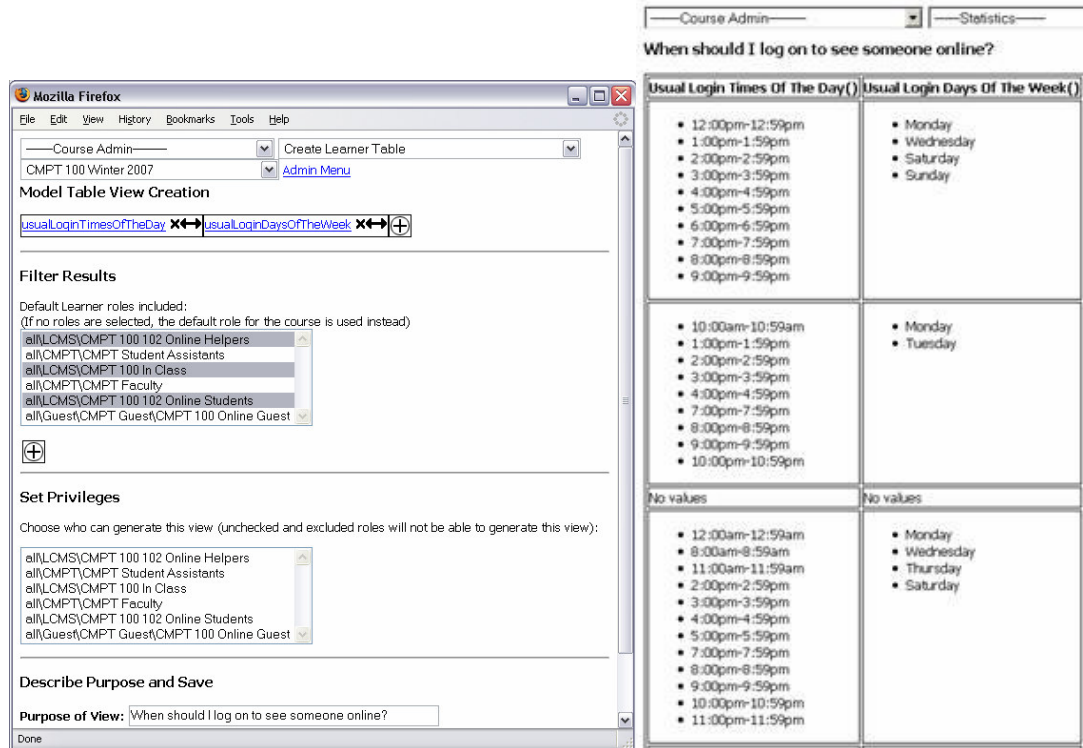


Figure 3-20: (a) When should I log on to see someone online (b) (Partial) Generated results of the view

Figure 3-21(a) is also a view that tells the instructor what time of day the learners log on, but the created view is in a different format than the table view in the previous example. Figure 3-21(b) is a table of log in times – each cell in the table represents a day/time, and the cells of the table highlighted in yellow are times the student had logged on to the system. By scanning vertically, the instructor is able to quickly identify which days and times are common log in times. By scanning horizontally, the instructor is able to quickly identify for particular learners which days and times the student logged on.

[illegible]

43

Figure 3-22(a) shows the complete query discussed earlier in the chapter (“is there anyone falling behind”), and the results of the query when generated. In this case, the instructor is interested in identifying students who may be falling behind in the online material – in this case, falling behind means not viewing all three lecture videos for the course. The results of the figure (b) show that at the time the view was generated 16 learners had not viewed the lecture videos, and that several had not used the online materials at all.

Course Admin Create Learner Table CMPT 100 Winter 2007 Admin Menu

Model Table View Creation

userid x name x timeSpentOnContentAndChildren(CMPT 100 Winter 2007) x

Filter Results

Default Learner roles included:  
(if no roles are selected, the default role for the course is used instead)

all/LCMS/CMPT 100 102 Online Helpers  
all/CMPT/CMPT Student Assistants  
all/LCMS/CMPT 100 In Class  
all/CMPT/CMPT Faculty  
all/LCMS/CMPT 100 102 Online Students  
all/Guest/CMPT Guest/CMPT 100 Online Guest

Filter In: hasRole(all/CMPT/CMPT 100/CMPT 100 Section 04) x  
Filter Out: viewed/Lecture Video: XHTML x  
Filter Out: viewed/Lecture Video: The Web x  
Filter Out: viewed/Lecture Video: Human Computer Interaction x

Set Privileges

Choose who can generate this view (unchecked and excluded roles will not be able to generate this view):

all/LCMS/CMPT 100 102 Online Helpers  
all/CMPT/CMPT Student Assistants  
all/LCMS/CMPT 100 In Class  
all/CMPT/CMPT Faculty  
all/LCMS/CMPT 100 102 Online Students  
all/Guest/CMPT Guest/CMPT 100 Online Guest

Describe Purpose and Save

Purpose of View: Is there anyone in SECTION 04 who is falling behind in  
☒ Overwrite existing view with this one  
☐ Keep existing view, create new view  
 Save View

Course Admin Statistics CMPT 100 Winter 2007 Admin

Menu

Is there anyone in SECTION 04 who is falling behind in the material?

userid()	Name()	Time Spent On Content And Children(CMPT 100 Winter 2007)
		6 hours 2 minutes 9 seconds
		3 hours 18 minutes 35 seconds
		0 seconds
		6 hours 13 minutes 31 seconds
		0 seconds
		8 hours 29 minutes 38 seconds
		4 hours 1 minute 22 seconds
		4 hours 3 minutes 50 seconds
		31 minutes 37 seconds
		2 hours 15 minutes 6 seconds
		0 seconds
		2 hours 16 minutes 50 seconds
		46 minutes 32 seconds
		0 seconds
		3 hours 3 minutes 6 seconds
		0 seconds

16 learners in table  
View generated in less than 1 second

(names and user ids omitted for privacy)

Figure 3-22: (a) Is there anyone falling behind (b) Generated results of the view

Figure 3-23(a) shows a different kind of query, where the instructor would like to know if any of the learners use the provided course CD. The generated results in (b) show that no one has set the preference to run from CD (a preference within iHelp Courses), meaning that they have defaulted to viewing the videos through the Internet (the choice by default if this preference has not been set).

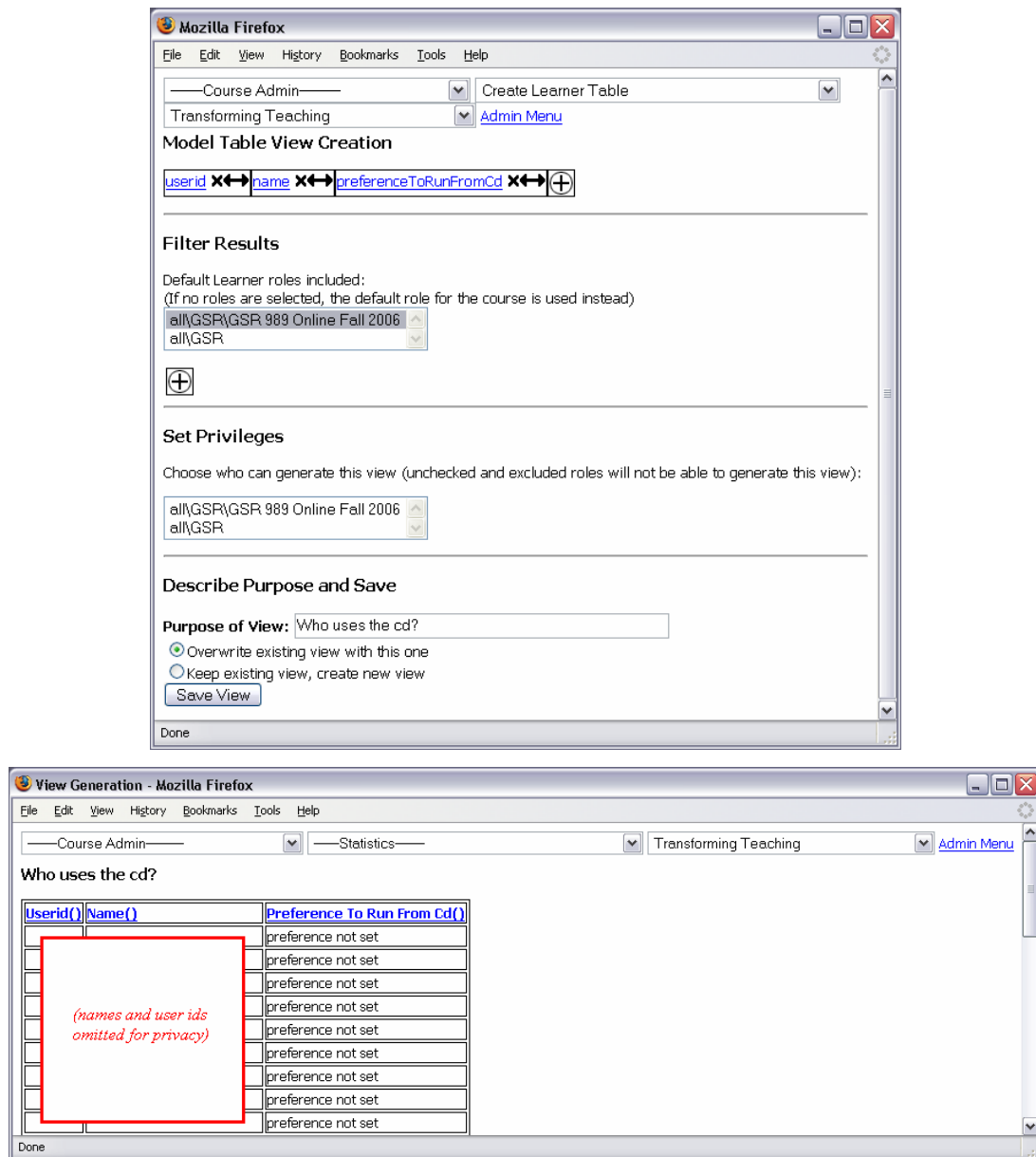


Figure 3-23: (a) Who uses the cd (b) (Partially) Generated results of the view



Figure 3-24 shows a view created to determine if learners are actively viewing content by showing the time the learners have spent on each of the modules in the course. The results generated show the learners have varying levels of activity in the course, but the amount of time each learner has spent on the material appears to be relatively constant from one module to the next.

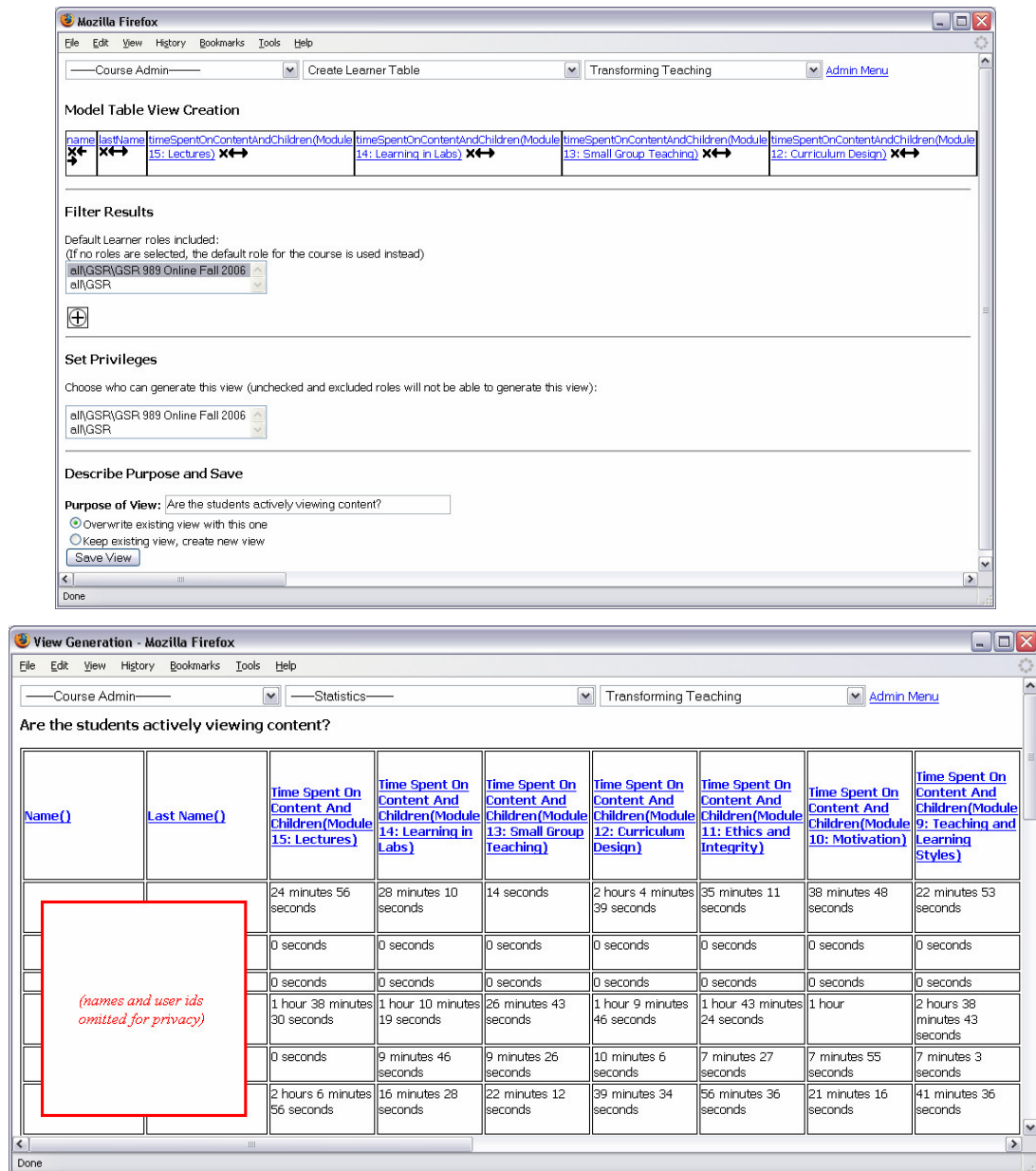


Figure 3-24: (a) Are the students actively viewing content (b) (Partially) Generated results of the view

Figure 3-25 is a query taken from a course in which participation formed part of the learner's grade in the course. Part of the mark was based on the number of discussion messages the learner had posted, and part of the mark was based on the number of discussions the learner had started in each module. The learner model shows both. Interesting trends can be observed from the learner model by scanning the rows of information – some of the learners participated often, some only participated in particular modules, and others didn't participate at all (although these could have been people who had dropped the course). The instructor for this class was interested in not only how many postings the learners made in the entire category (where discussion categories were set up for each module in the course), but also the number of discussion threads the learner had started for that week, meaning discussions the learner had in essence started. Reviewing the results of the view also gives an indication of the overall activity of the group, and which modules/topics were controversial or discussed heavily in the course forums.

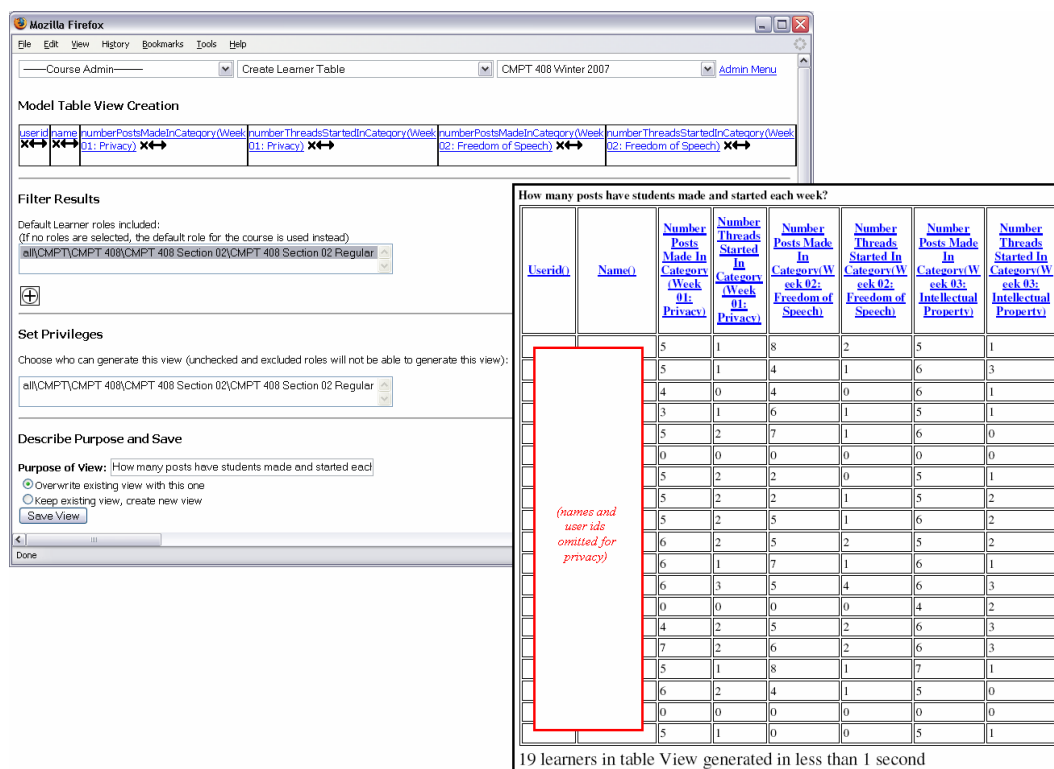


Figure 3-25: (a) Participation marks by postings made and started (b) (Partially) Generated results of the view



The learner model in Figure 3-26(a) also shows a view created to describe participation. However, in this course the instructor has determined participation in the course is based on different parameters: the number of posts the learner has made, and the number of posts the learner has read in the course. The first column in the table (participationCMPT100OnlineOutof10) is a system-calculated measurement of participation determined by the instructor and coded into the Query Tool using the computation in equation 3.1. This computation is a process of the Query Tool, made available to the instructor for this one course. Initially, the design of the Query Tool included a mechanism for instructors to specify such computations themselves using raw data captured by the system to build other processes, and the system does have the underlying ability for such additions. However, an interface flexible enough to enable this capability proved difficult to create, so instructor-created processes were left out of the system (though in retrospect weighted sums or simple averages could be very useful, and not as difficult to incorporate). Pre-programmed methods based on instructor requirements were included in the system instead. Figure 3-26(b) shows a generation of the view on data from a test course.

$$\text{Participation} = 0.7 \times P_m + 0.3 P_r \quad (3.1)$$

where  $P_m$  = the percentage of posts the learner made (as computed by equation 3.2) to a maximum of 100%

and  $P_r$  = the percentage of posts the learner read (as computed by equation 3.3) to a maximum of 100%

$$P_m = N_m / A_m \times 100\% \quad (3.2)$$

where  $P_m$  = the percentage of posts the learner made

and  $N_m$  = the number of posts the learner made

and  $A_m$  = the average number of posts made by peers

$$P_r = N_r / N_t \times 100\% \quad (3.3)$$

where  $P_r$  = the percentage of posts the learner read

and  $N_r$  = the number of posts the learner read

and  $N_t$  = the total number of posts

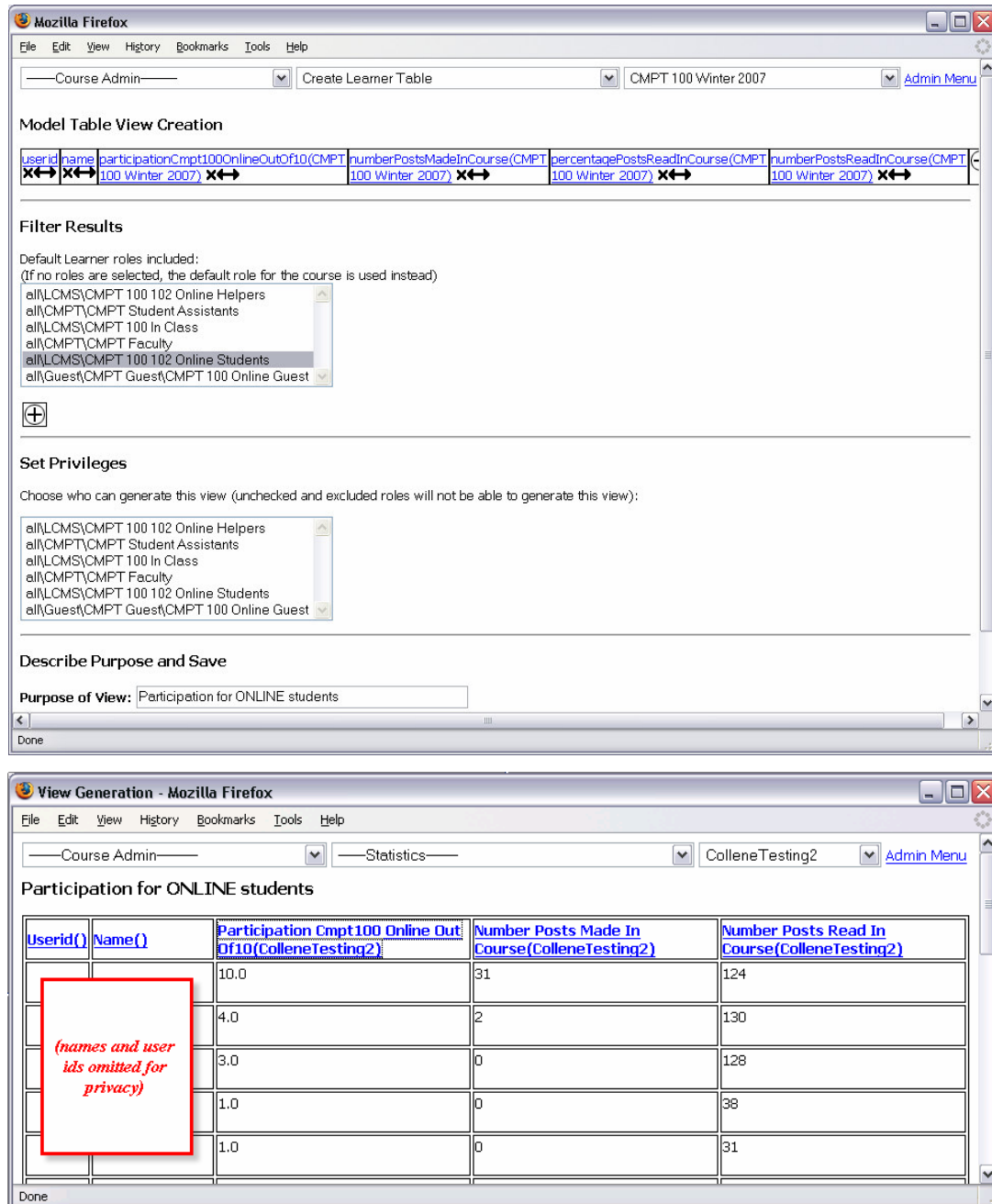


Figure 3-26: (a) Participation marks by postings made and read (b) Results of the view when generated

### 3.1.6. Observations from Real Views for Instructors

Instructors have diverse needs when viewing information about their learners. Thus, a query tool to be used by instructors needs to be flexible yet functional. The figures and discussion above show that underlying raw data can be combined in interesting ways to fulfill a purpose the instructor has (the very heart of active open learner modelling). By giving the instructors a way to interactively define what their purpose is, instructors are not limited in what types of information they can obtain from the system. For example, two different instructors decided on two different methods of calculating participation, but both purposes were enabled by the system.

Characteristics when displayed do not always follow the same format. For example, when the characteristic is “name”, the generated result is just text. When the characteristic is “usual log in days of the week”, the generated result is a list of days. When the characteristic is “access times in last week”, the generated result is a chart of dates the learner has accessed the course in the last week. If the generated information in the column is sortable (for example, names and dates are sortable, but lists of items and tables are not), a link appears on the top of the column to sort the data, either in ascending or descending order. The display changes based on what has been calculated by the learner model computation.

External activities typically drive the purposes – the instructors have questions about their course they wish answered and other actions outside the system (such as communicating with the students) are taken to react to the information presented in the system. Potentially, the system can be used to automate these actions to automatically respond to certain actions in the system, though this is not currently implemented in the current system. For example, learners may be sent emails automatically after a certain date if they have not yet logged on, or extra exercise pages assigned to those who are identified as requiring the practice.

While only one type of view has been created using the system (a table view), other types of views are also possible, but have not yet been implemented in the current system. For example, a tree view could be created which uses the course content organized in a tree structure to display learner characteristics related to course content in another way (see Figure 3-27). In this view, the instructor could see in one structure how the learner is doing in each topic of the course, and see how material at a lower leaf level in the tree affects the coarser grained items. Another view is a time-elapsd view where progress is shown with a timeline (see Figure 3-28). Each of these views provides a useful view of information in the course, and alternative views can be created as necessary using the same system design.

	Userid: abc123 Name:		Userid: abc123 Name:	
	TimeSpent	Participation	TimeSpent	Participation
[-] CMPT 100 Winter 2007	17 minutes	8/10	18 seconds	1/10
[+] Getting Started	38 minutes	6/10	49 minutes	5/10
[-] Course Content	1 minute		12 seconds	
[+] Evolution of Computers	2 hours	5/10	17 minutes	10/10
[+] Software Systems	3.5 hours	5/10	5 hours	4/10
[-] Networking and the Internet	30 seconds	5/10	0 minutes	0/10
• Lecture Video: Networking Part 1	42 minutes		0 minutes	
• Lecture Video: Networking Part 2	29 minutes		0 minutes	
• Lecture Video: Networking Part 3	8 minutes		13 minutes	
• Lecture Video: Networking Part 4	0 minutes		7 minutes	
• Lecture Notes: 3 to a page	0 minutes		0 minutes	
• Lecture Notes: 6 to a page	0 minutes		0 minutes	
[+] What Are Networks	34 minutes		0 minutes	

Figure 3-27: Example tree view

Userid: abc123	Time spent	47 minutes	13 minutes	8 minutes
Name:	Participation	3/10	7/10	6/10
Userid: abc123	Time spent	16 minutes	0 minutes	2 minutes
Name:	Participation	1/10	2/10	8/10
		Sept 1/06	Oct 1/06	Nov 1/06
				Dec 1/06

Figure 3-28: Example timeline view

All of the queries are currently generated in real-time – the calculations are not cached or stored or computed offline. The “expected response time” for each view is computed from previously generated views and updated each time the query is generated to give the users an approximation of how long it will take for the results

to return. In general, most views return results in a few seconds – larger classes with many learners require more time, as do views with complicated calculations. For most of the queries generated for the courses described in this chapter, this was satisfactory. However, for new views (the timeline view in Figure 3-28 in particular) or longer computations, this may not hold true. Fragmenting and storing partial calculations for faster computations is also possible using this system design. Each characteristic/column in the view is calculated separately in the system design (see Appendix C). These partial results could be cached and used in later computations if the results are still appropriate. This introduces complexity into the system design, but the advantages of using this approach may justify the added complexity. While an initial examination of using cached results has been done and determined possible, in general this approach was decided to be a future direction of this research.

Instructors are always able to see all information about the learners in their course (or at least information provided by the system as appropriate for instructors to see). However, learners may not want all information to be revealed to their instructor. There is a balance to be maintained here – instructors need the ability to accurately assess their students' progress, versus the learner's concerns of privacy. Determining what information is appropriate to display to instructors, or implementing a method to enable the learner to decide is a future direction for this research.

The raw data from the users can potentially be used in more complicated calculations to determine further characteristics of the learners (as the participation calculation shows). In this case, the method of creating and generating the learner models is the same for the instructor, but the process is extendable as the instructor requires. It is possible to enable the instructor to create their own characteristics from underlying raw data (for example, creating their own participation characteristic from the appropriate raw data as suits their situation). The system design (see Appendix C) encourages this ability as the computations or

characteristics map to actual methods in the code, and Java Reflection [SUN2007] can create such methods at run-time as the instructor creates them. However, designing a suitable and secure interface proves to be the most difficult aspect of this function, as it is difficult to create an interface with the maximum functionality without also creating a full programming environment. Many instructors using the course delivery engine are not computer programmers. The problem of security arises as well – with maximum ability and openness using a programming environment capable of doing “anything” comes the ability for security exploitation. Therefore, this is another area for future research.

### 3.1.7. Real Views and Sample Results for Students

This section describes real queries created by instructors for their students to view for the same courses in the previous sections, and a few sample queries from a test system to show capabilities not used in the production system used by instructors. Not all of the courses opened the views to students in the course, for various reasons, discussed in Chapter 4. Even for the courses where the instructors did open the views to the learners, their reasons differed. Several of the more interesting queries opened to learners are presented here, with generated results. Note again that any identifying information is erased from the screenshots below, but in situations where the learner would see -BLOCKED-, this remains on the screenshot. In the generated views, if the learner appears on the views, all of their own information is displayed, and their row is highlighted in yellow to allow them to easily identify which information is about them.

In Figure 3-29, the purpose of the view is to tell the learner how they are doing in the course compared to their peers. In this case, the instructor has included the participation, percentage of posts read in the course, time spent on all the content to date, and the last page viewed in the course, all as comparisons the learner may use. Motivations for students may differ as they complete a course or participate in class discussions and other collaborative activities with peers ([GREER2001] [BRETZKE2003]). By giving students multiple calculations of “how am I doing”

on the same view, the instructor's goal in this case may be to provide as many motivating elements on one view as possible. For example, the learner may want to compare his/her participation to see if it matches that of other peers in the course, either to change his/her participation mark or for additional social recognition in the community [CHENG2005]. The learner may want to see how long their peers have spent on content to see what others in the course thought was important to spend their time on. Identifying where others are currently in the course is also motivating – “am I behind or ahead of my peers?” [KAY1997].

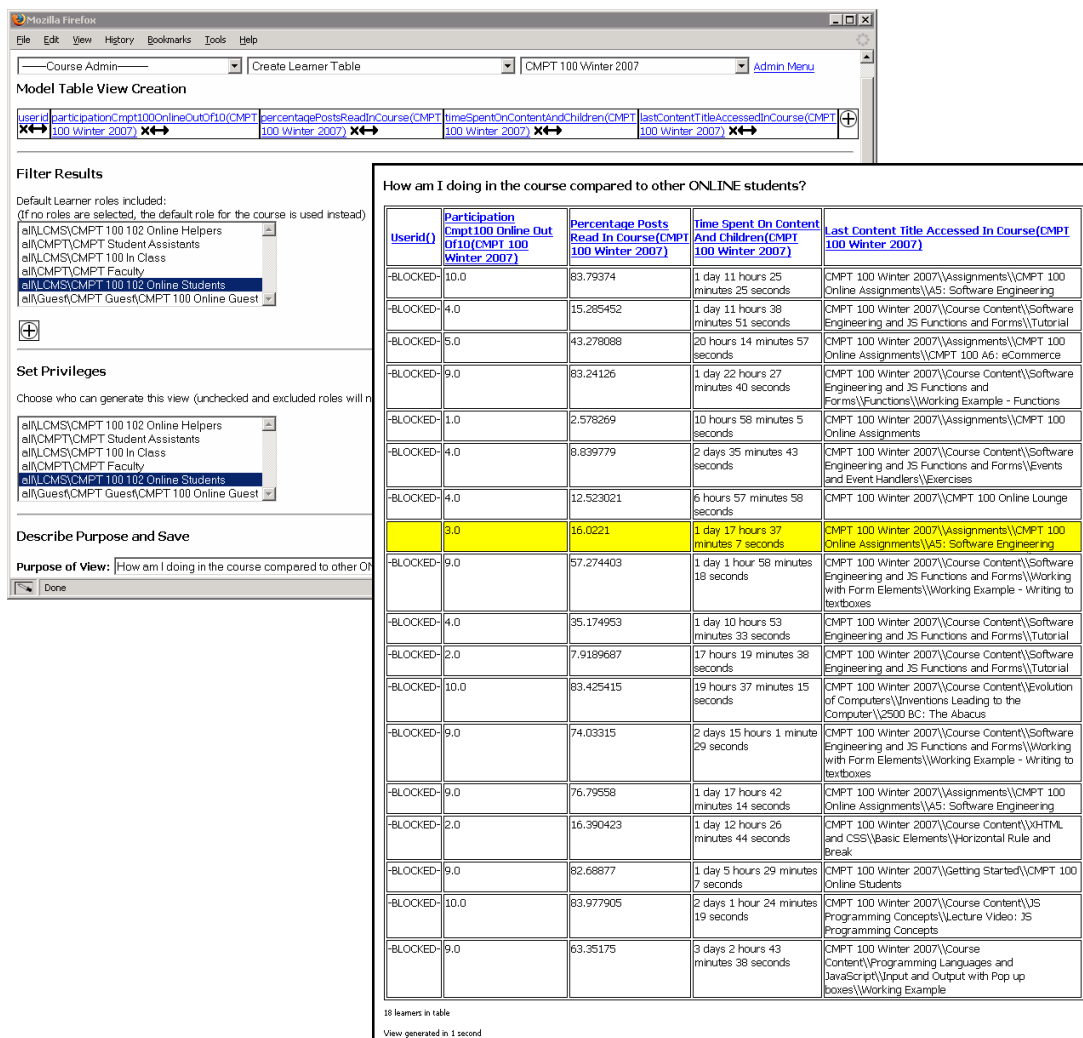


Figure 3-29: (a) How am I doing (b) Generated results of the view

A few instructors used the filters provided by the system to specify a subset of learners returned in the view; however, these views were not subsequently opened to the learners in the course. One of the features of the Query Tool is the ability to block filters as well as characteristics, in case the filter being applied is able to limit the results so much as to specifically allow identification of a particular learner. In the case of a blocked filter, the filter will simply not be applied, in this case providing more learners in the view, but still allowing some level of comparison to be completed. This uses the philosophy “better to have too much than no information at all”. Figure 3-30 shows a view created in a test system to describe this feature of the Query Tool. In this case, a view was created to display contact information for those with guest accounts to access the course. Guest accounts in this case have a userid starting with “guest”. Figure 3-31 shows the view generated by an instructor, who is able to access everything in the course. Figure 3-32 shows the view generated by a learner in the course. In Figure 3-32, there is a warning at the bottom showing how the filter was not applied, and there are many more learners on this page than the page generated for the instructor.

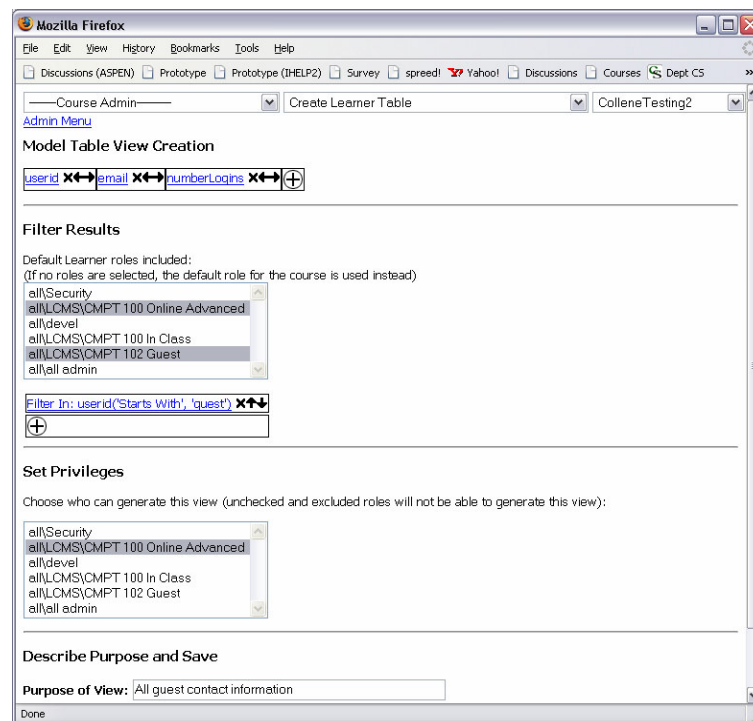


Figure 3-30: View created to list contact information for guests



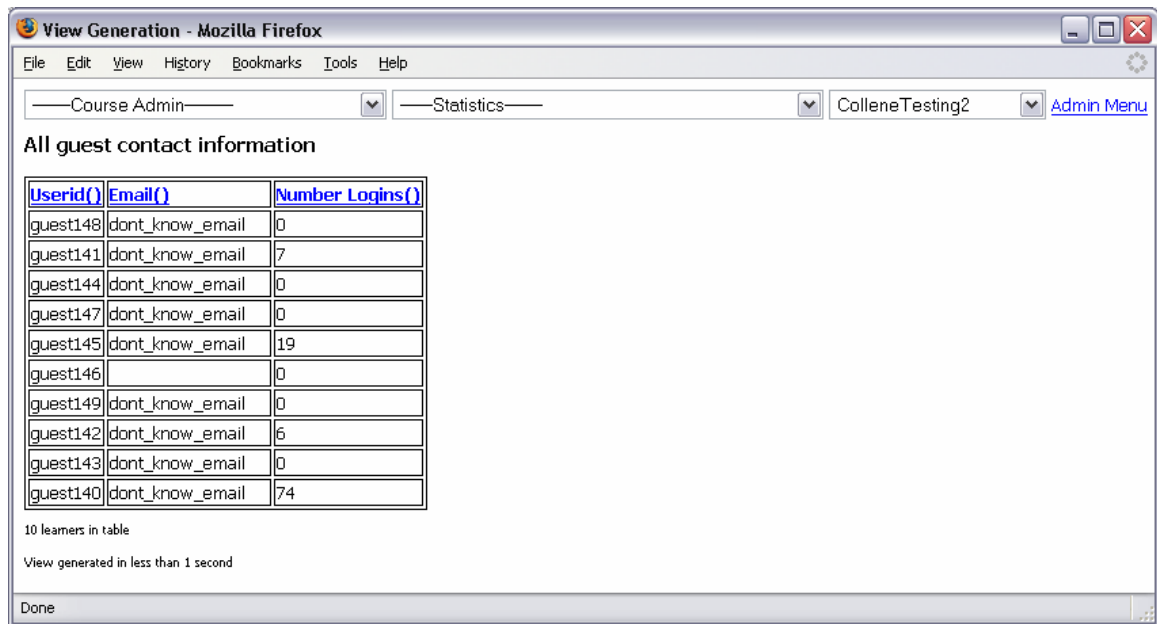


Figure 3-31: View generated by instructor

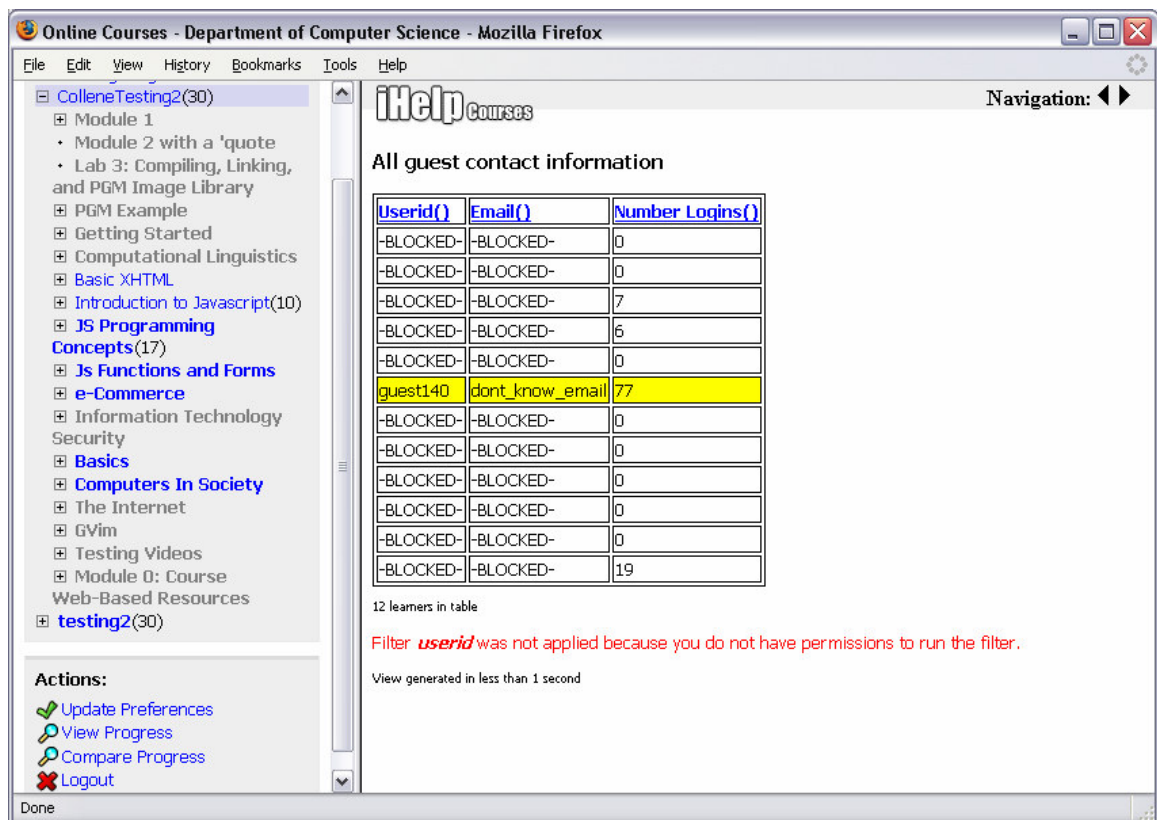


Figure 3-32: View generated by learner

### 3.1.8. Observations from Real Views for Students

Students are not presented with a set of different views from the instructors – instructors and learners in the system can generate the same views (and in fact in the underlying system design, the same code is called to generate the information). However, the information presented on the views depends on who the user is – learners may be blocked from seeing some information, or may see some of the information differently. Thus, who the user is affects *what* is displayed to them about themselves and others in the course and *how* it is displayed. In active modelling terms (from Chapter 2), the subject doing the modelling affects the contents of the learner model.

One of the important features of the system is the privacy protection enabled by blocking characteristics and filters from those who should not be able to use them. Opening learner models to other learners in the course also opens the proverbial “can of worms”, since privacy is important to consider when displaying sensitive personal information such as course marks or learning progress. In the examples in the previous section, learners were blocked from seeing certain information about other learners, and this was decided to be appropriate by the instructor based on the learner’s role in the course. Even though some information was blocked, the system still made the “best effort” to deliver useful information to the learner. In most cases, anonymous or statistical characteristics still enable the learner to compare their progress to others in the course. In the case of filters (which limit the number of learners returned in the view), if the filter was blocked, there were just more learners returned instead of a more limited view. The “best” information was not necessarily presented (since the filter did not limit the results to only those described by the view purpose), but the learner was still able to view the characteristics for all the learners, which provides some useful information to the viewer.

Learners should not be able to individually identify other learners by using information calculated by the system, as this would violate privacy. The privacy

protection in the Query Tool does provide a high degree of protection, but it is very difficult to implement to cover (or even identify!) every possible situation. External influences, using information from different sources, and combining information may allow a learner (either being purposefully malicious or just by accident) to deduce the identity of another learner in the course. For example, consider the case where an instructor decides to open the characteristic “number of discussion messages in a category” to the learners in the course. The discussion boards allow learners to adopt “aliases” to offer pseudonymity in the class discussion, but often the learners in a course have defaulted to use their first name or last name as their alias. Assume as well that for the purpose of the view, the instructor has added further information to the view such as a quiz mark. A learner (LearnerA) may generate the view to see the number of messages in the discussion forum at time1. In the meantime, LearnerB posts a new message to the discussion forum using their default alias. Shortly after at time2, LearnerA may generate the view again, and observes that one of their peers’ number of messages just increased by one. By going to the discussion boards and seeing that LearnerB posted a message between time1 and time2, LearnerA may then learn the individual identity of LearnerB and therefore LearnerB’s quiz mark that is also included on the view, even though the instructor blocked names from being displayed.

A further enhancement to the privacy protection to be considered as future work is to allow the learner to only open up information to trusted groups or individuals [BULL2005]. In this case, the learner could assign levels of access to information in the system based on the groups they create, or default groups such as instructors, teaching assistants, and fellow peers. For example, they might allow friends to see some information calculated about them that other peers in the class may not have access to. Or, they might allow instructors to view some information but not their peers. However, paranoia could set in on some users, who may then block all characteristics from everyone, as the system may not be trusted to display only appropriate results to other users.

Another privacy concern (initially described in Chapter 2) is the situation where a viewer (LearnerA) uses characteristics on the view to (correctly or incorrectly) deduce further characteristics about another learner (LearnerB) which were not generated by the system. For example, CharacteristicA and CharacteristicB are displayed on a view, and LearnerA uses them to deduce CharacteristicC about LearnerB. This situation is further complicated because LearnerB may have allowed CharacteristicA and CharacteristicB to be displayed to LearnerA, but did not know CharacteristicC could be deduced, and may not want CharacteristicC to be known to someone else. As an example, consider a situation where the instructor has created a view which displays the time learners spent on a particular module in the course and the number of quiz attempts on the module's quiz. LearnerA may see that LearnerB spent a long time on the module and attempted the quiz many times, and may jump to the conclusion that LearnerB performed poorly on the module's assignment (an extreme deduction, but possible). LearnerB may not want this information to be made public to LearnerA, or at least not without additional information to clarify the view (perhaps LearnerB was studying the module in more depth for work commitments, so wanted to make sure he/she understood the material fully). To help with this type of situation, LearnerB could be given the ability to justify or disagree with the learner model or add extra information to it [DIMITROVA2003b]. At least, LearnerB should be able to see what information is being generated and displayed to LearnerA, which the Query Tool currently allows.

### 3.1.9. Summary of the Query Tool

This chapter introduced the active open Query Tool (also known as the Compare Progress feature of iHelp Courses) developed to enable reflection for instructors and learners in online courses. The features of the Query Tool were described in light of the actual courses using the Query Tool. Recall the goals of the Query Tool introduced at the beginning of the chapter, and consider how the presented examples discussed in the chapter fulfill these goals:

**1. Allow instructors and learners to obtain a sense of where learners are in the course and how well they are doing**

This is achieved by opening the learner modelling information (such as participation and level of progress in the course) to the user. The chapter presented examples of real queries created by instructors to gauge progress in each of the different types of courses. Some of these same queries were also available to learners to then compare themselves to others in the course to reflect on their progress.

**2. Give the user more control of the modelling process by being involved in the modelling process**

The instructor specifies what it means to “do well” in the course, making the Query Tool highly adaptable to different situations. The instructor chooses characteristics from raw data stored in the iHelp systems, and limits the learners displayed in the model using filters to target the model even further. While at this stage of the research learners are unable to create views to compare themselves to others with characteristics they wish (this is considered a future direction), the learners choose which views to generate to ask questions of the system, giving the learners limited involvement in the modelling process.

**3. Create an interactive interface to encourage the user to dig deeper into the model**

The descriptive purposes provided by the instructor give the learners specific questions to pose of the system as an intuitive draw to seek further information. Sorting features on columns and multiple display formats for the characteristics provide an interactive interface once the views are generated.

**4. Provide a privacy mechanism flexible enough for the learner's privacy to be protected, yet keep a level of usability of the information presented to the viewer**

The privacy protection used by the Query Tool blocks characteristics and filters based on who is viewing the learner model. As such, compromises are made between the best results of the learner model and the privacy of learners in the course. Identifying information (such as names or user id's) is blocked by default by the system, but most information is kept open until the instructor chooses to block it. Blocking a few characteristics while opening as many as possible keeps the system usable to the viewer. Some filters may potentially limit the set of learners returned enough to allow individual learners to be identified by their peers, so filters, as well, can be blocked by instructors. In this case, viewers who are not able to run the filter see more learners than the filter, if applied, would return, providing more information to the learner, but not necessarily the "best" results that the filter would return.

When comparing the Query Tool to other existing system in content management systems (CMSs) (such as Moodle [MOODLE2007] or WebCT [WEBCT2007]), the system does provide several advantages. Using the Query Tool, an instructor is able to create a targeted view to meet his/her needs on one page – other systems provide similar information, but the information is scattered across several pages. Additionally, the Query Tool allows the views to be opened to learners, whereas other CMSs only provide information in their reporting tools to instructors. The data available to instructors through the Query Tool (provided by the iHelp Suite) is also quite extensive when compared to other systems.

### *3.2. Research Questions*

The Query Tool by design is quite open and flexible to try to meet the needs of a larger group of instructors than other similar tools provide. To examine the effectiveness of the Query Tool and to determine if the Compare Progress feature meets the goals outlined in this chapter, several specific questions are posed:

- Have instructors and learners used the Query Tool, and for what purposes?
- Are there features of the Query Tool that were not used?
- Does the Query Tool give instructors and learners an appropriate indication of how learners are performing in the course?
- Is the information provided by the system deemed accurate by instructors and learners?
- Did the behaviour of the learners and instructors change based on information displayed in the views?
- Is privacy protected by the mechanisms in place? Do learners feel their privacy is sufficiently protected?
- Considering the compromises of privacy versus usability discussed in the chapter, do users still feel the system is usable with the privacy protection in place?

These and other questions are explored in the next chapter.

## 4. STUDY AND RESULTS

The previous chapter described the Query Tool created to actively open learner models using the iHelp Courses system. This chapter outlines the research questions explored by the Query Tool and the process taken to study the effectiveness of the Query Tool in actual online and blended learning courses. In general, the Query Tool was analyzed in two ways: the usage of the Query Tool by instructors and learners to determine if and how the feature was used, and the effects of the Query Tool on the courses.

### *4.1. Overview of Study Procedure*

Four courses with online learners used the Query Tool. Two of the courses are entirely online courses, while two of the courses are blended learning courses. Three of the courses are Computer Science courses (two junior level, and one senior level), while one course is an informational course on university teaching practices. Each course had different reasons for creating learner models. Two of the courses used the views to establish the level of participation in the course, though each course had a different idea of what constituted participation. Several of the courses used the views to determine if students were keeping up with the online materials for the course, and to offer encouragement to those who were falling behind. One of the courses used the views to determine the effectiveness of some of the course materials. Only one of the courses opened the learner models to learners in the course.

To analyze the usage of the Query Tool, statistical analysis was completed on the use of the system across two terms and compared against usage data captured by the iHelp systems. The Query Tool captured when users created new views, edited



views, or generated views. Using the timestamps of these events and timestamps of events in the iHelp suite (such as page views, message board postings, and quiz attempts), interesting patterns can be found. However, while statistical analysis can provide interesting patterns and information, it is difficult to answer the question “*why* did the users act this way” without jumping to conclusions about the data. So, to understand “*why*”, questionnaires as part of a proof of concept study were then used to gather more qualitative responses from users in an attempt to find the effects of the Query Tool on teaching and learning and social interaction in the courses.

#### *4.2. Instructor Statistics, Survey, and Results*

Instructors from two courses (both courses had multiple instructors) were asked to complete the questionnaire, and two instructors actually submitted it. Results this small can not be considered statistically significant, but the suggestions and experiences from the questionnaire still provide unique insights into how the tool was used by instructors in their actual courses as a proof of concept study.

I was also an instructor for one of the courses that used the Query Tool (the course which opened the learner models to learners, discussed in the following section). However, I did not include my own data in the following analysis of instructor usage and did not complete the questionnaire.

##### *4.2.1. Query Tool Usage and Usefulness*

In total, four instructors directly used the Query Tool (several of the courses had multiple instructors), and another instructor used the information from the Query Tool indirectly through a teaching assistant who used the system (this is summarized in Table 4-1). A few instructors used the system initially at the beginning of the course, but stopped mid-way through the term. One instructor used the system periodically throughout the term. Another instructor only used the system at the end of the term. Overall, the tool was not used extensively by the instructors.

Course	Term	Type of Course	Number of Instructors	Number of questionnaires completed by instructors	Number of students	Number of questionnaires completed by students	How Query Tool was used
CMPT 100 regular sections	Winter 2007	blended	2	1	118	n/a	One instructor used the query tool to create and view queries to determine if students were using the online materials for the course. Only the instructor viewed the learner models.
Transforming Teaching	Winter 2007	online	3	1	30	n/a	Three instructors used the query tool to create and view queries to determine how the students were doing in the course. Only the instructors viewed the learner models.
CMPT 408	Winter 2007	face to face with online discussion forums	1	0	19	n/a	The course's teaching assistant viewed the queries and sent information to the instructor. Only the teaching assistant viewed the learner models.
CMPT 100 online sections	Fall 2007 and Winter 2007 (two offerings)	online	1	0	16	5	I was the instructor for this course, and used the system frequently to gauge the learners' progress through the term. Learner models were opened to students.

Table 4-1: Summary of Courses using the Query Tool

The tables Table 4-2 to Table 4-4 show the views created by instructors in three courses, and short descriptions of what the view was created to do. Some of the views were created by instructors (as the tables indicate), but a few of the views were created by the system administrator (myself) in response to requests from instructors. The instructors all created very different views with varied purposes. The instructor of the CMPT 100 regular section (Table 4-2) used the Query Tool to determine if learners were using the online course materials effectively. The instructors in Transforming Teaching (Table 4-3) used the system to monitor the progress of the learners through the course. The instructor in CMPT 408 (Table 4-4) used the system to give credit to learners for participation in the class discussion forums. Each instructor tended to favor one particular view over the others. The times generated in these tables include only the times course instructors or teaching assistants generated the view – usage by system administrators was not included in these tables.

Figure 4-1 shows timelines created for each course to show instructor usage of the Query Tool throughout their respective terms. In the timelines for Course 2 (Transforming Teaching from Table 4-1) and Course 3 (CMPT 408 from Table 4-1), use by the system administrator is also included, as the system administrator provided answers to instructor requests about these courses using information provided by the Query Tool. The timelines again show that the instructors used the Query Tool sporadically throughout the term, though they tended to generate several queries in one session. On the questionnaire, both instructors who responded indicated they used the Query Tool “occasionally” throughout the term.

View Purpose (By Instructor)	Created By	Times Used	Summary of View
Last log in times and time spent on course for ONLINE students	Admin	2	To find when the learners in the online section of the course last logged on, and how much time these students spent on the course material
Which INCLASS students are using the material and how much?	Admin	2	To find the names of the learners in the other sections of the course who used the online materials, and how long these students had spent on the online course materials
Has anyone watched the lecture video?	Admin	2	To find the names of the learners who watched the lecture video titled "Evolution of Computers"
How am I doing in the course compared to other ONLINE students?	Admin	4	To open to learners to tell them their participation in the course, along with the time they have spent and the last content title accessed. Learners can see their peers' information as well for comparison
What is my participation mark?	Admin	1	To show only the learner viewing the page his/her participation mark
Who has watched the Evolution of Computers video?	Admin	1	To show a list of learners who watched the lecture video titled Evolution of Computers. Also includes an indication of whether the learner is an online or on campus student
INCLASS general information - usage, login times	Admin	1	Shows the name, the last content title accessed, the first time the learner logged in, and the last time the learner logged in for only on campus students who had accessed the course during that term
who inclass saw videos module 2	Instructor	1	Shows a list of learners and the time spent on the lecture videos from the second module. Only included on campus students who had accessed the course during that term
time spent on network videos (inclass)	Instructor	2	Shows a list of learners and the time spent on the lecture videos from the third module for only on campus students who had accessed the course during that term
time spent on hci and web videos	Instructor	13	Shows a list of learners and the time spent on the lecture videos from the fourth module for only on campus students who had accessed the course during that term
viewed web and hci video	Instructor	5	Shows a list of learners and whether they viewed the pages of the lecture videos from the fifth module for only on campus students who had logged on to the system since the beginning of term

Table 4-2: Views created for Course 1 (CMPT 100 regular section)

View Purpose (By Instructor)	Created By	Times Used	Summary of View
General Statistics for Course	Admin	19	Shows a list of learners with the learner's user id, name, email, last log in date, total number of log ins, time spent on content, number of posts made and read in the course
Usual log in times of the day	Instructor	2	Shows a list of learners and the time of the day they usually log in to the course
Access times in the last week	Instructor	3	Shows a list of learners with a table of the days and times they had logged on in the last week
Contact information for students who have not yet logged on	Instructor	3	Shows a list of learners, their name, and their email. Only includes learners who haven't logged on to the course yet
Who uses the cd?	Admin	1	Shows the user id, name, and the preference to run from cd for the entire course
Overall time spent on the course	Admin	2	Shows the user id, name, and the total amount of time the learner has spent on the content
Time spent on content for the weeks 0-4	Admin	1	Shows the user id, name, and time spent on each of the content for the first five weeks of the course
Feb. 28 Module 9	Instructor	1	Shows the user id and the last time the learner left the content page Module 9: Teaching and Learning Styles
test	Instructor	1	Shows the user id, time spent on the course's title page, and total time spent on the course content
List of all students	Admin	1	Shows the user id and name of all the learners in the course
When should I log on to see someone online?	Instructor	5	Shows the user id, name, last name only, usual day of the week the learner logs in, usual time of the day the learner logs in, and all of the access times in the last week for all the learners
Is there anyone who is falling behind in the material?	Instructor	1	Shows the name, last name only, number of posts read and made in the course, number of log ins, last log in date, number of log ins in the last week, percentage of posts read in the course, last date content was accessed in the course, and the total time spent on content and children
Are there students actively viewing content?	Instructor	2	The name, only last name, and total time spent on content in the course
Are there students actively viewing content?	Instructor	2	The name, userid, only last name, and total time spent on each module in the course
end of course stats	Instructor	1	The name, email, last log in date, all the times the learners accessed material in the course, and number of posts made in the course
final course stats	Instructor	1	The name, email, number of posts made in the course, last time the learner accessed material in the course, and last log in date

Table 4-3: Views created for Course 2 (Transforming Teaching)

View Purpose (By Instructor)	Created By	Times Used	Summary of View
Requirements for Introduction Week	TA	0	Shows the user id, name, number of threads the learner has started in the discussion forum for the first week, the total number of posts in the discussion forum for the first week, and the number of posts he/she has read in the discussion forum for the first week
Requirements for Introduction Week	TA	2	Shows the user id, name, number of threads the learner has started in the discussion forum for the second week, the total number of posts in the discussion forum for the second week, and the number of posts he/she has read in the discussion forum for the second week
Requirements for Privacy Week	TA	2	(Same as previous)
Requirements for Privacy Week	TA	2	(Same as previous)
How many posts have students made and started each week?	Admin	4	Shows the userid and name of each learner in the course, followed by the number of threads the learner started in the discussion forum for each week, and the total number of posts the learner made in the discussion forum for each week

Table 4-4: Views created for Course 3 (CMPT 408)

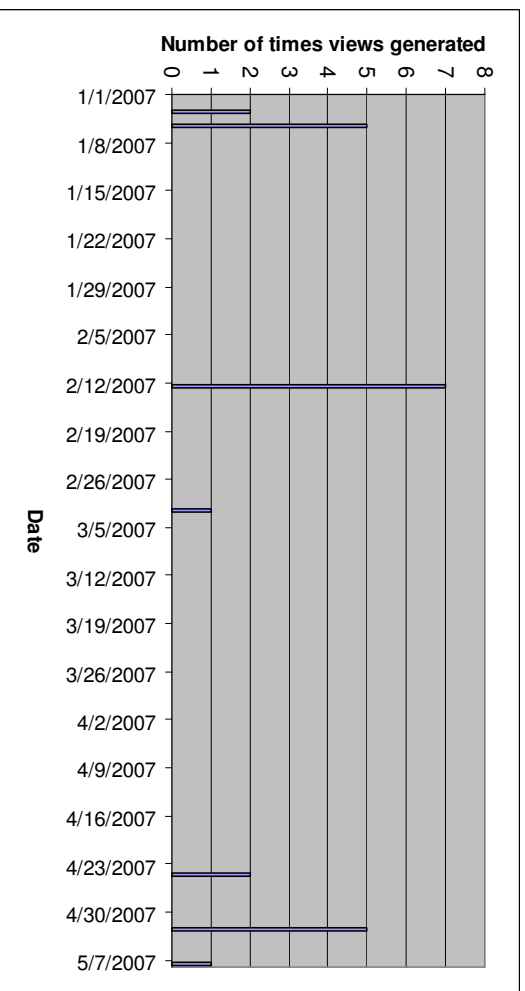
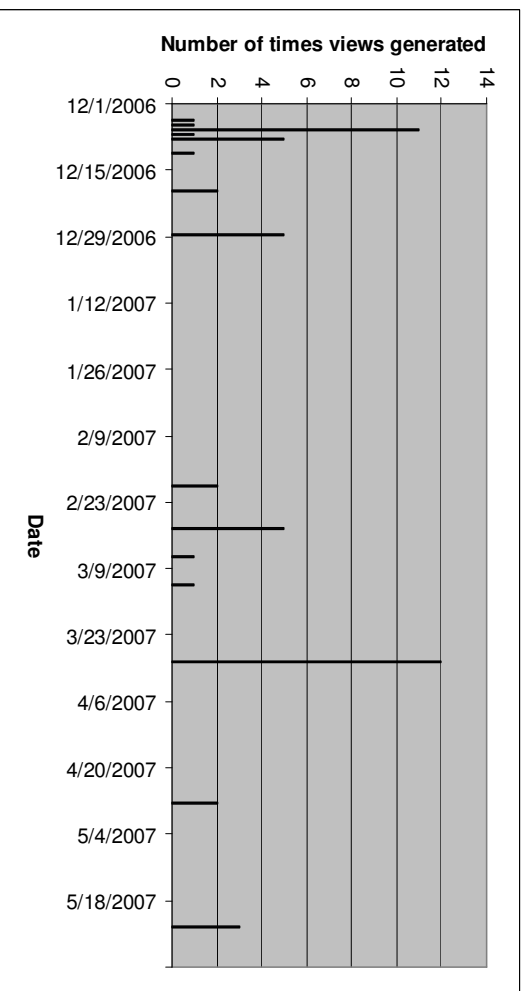
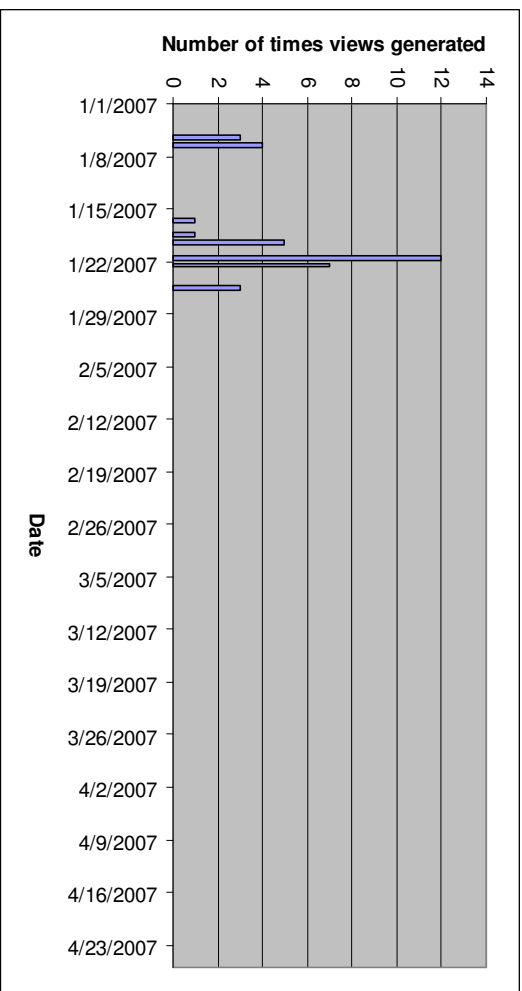


Figure 4-1: Number of times instructors generated views in (a) Course 1 (b) Course 2 (c) Course 3

Although the instructors did not use the Query Tool extensively in their courses, the instructors who responded to the questionnaire indicated they appreciated the abilities the system had. As one of the positive aspects of the system, one instructor commented it “*provided useful information*”, while the other instructor appreciated the comprehensive data that was gathered and the ability to “*keep tabs on my students right to the minute*”. Both responded “strongly agree” to the statement on the questionnaire “I am able to find out valuable information about learners in my course”.

The instructors appeared to use the Query Tool for specific purposes in their courses, which affected their responses on the questionnaires. Instructor 1 (Figure 4-2a), who instructed a fully online course (Transforming Teaching from Table 4-1), used the query tool to identify learners falling behind and contact them if necessary, monitor participation, monitor and evaluate the use of the course materials, and determine how to adjust his/her schedule to be online at the same time as the learners. Instructor 2 (Figure 4-2b), who instructed a blended learning course (CMPT 100 regular sections from Table 4-1), used the query tool for similar reasons, but did not use the system to adjust his/her schedule, and indicated the goal to open the views to learners in the course. However, the views were not actually opened to learners in the course, for reasons the questionnaire results did not capture. This could have been partially due to the interface design of the view creation process (this and other problems with the interface are discussed later in the chapter).

When the instructors were asked if the views they created fulfilled the original purpose they had, Instructor 1 said “strongly agree”, while Instructor 2 disagreed. Instructor 1 did not encounter a situation where the system did not support his/her goals in using the Query Tool, whereas Instructor 2 did encounter this situation. When asked to clarify, Instructor 2 described the situation as not being able to obtain detailed information about the student interaction with the course material in which he/she was interested. This course offered many online videos. Limitations of the



course delivery system at the time meant it was impossible to capture how long the learners watched the videos, though it was possible to determine if the learners had viewed the page with the videos, which does not even guarantee the learners even watched the videos. So, in this situation, the data captured and made available by the Query Tool limited the usefulness of the system.

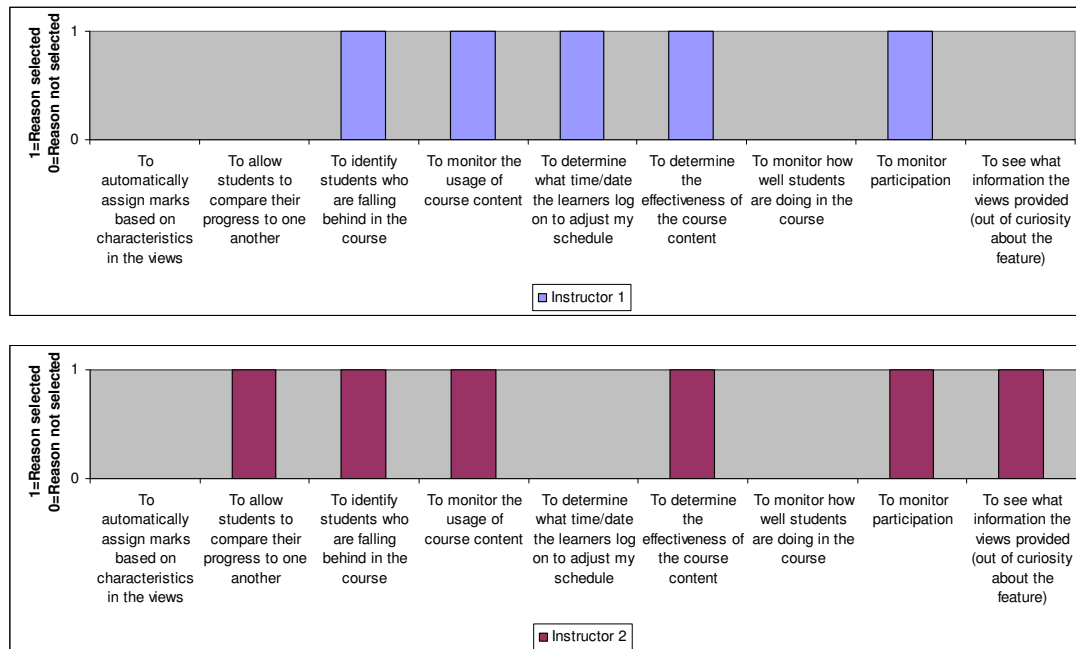


Figure 4-2: Reasons instructors used the Query Tool (a) Instructor 1 (b) Instructor 2

To analyze the appropriateness of the information that was captured by the delivery system and made available to instructors through the Query Tool, the instructors were asked to rate the most common characteristics in the system as important, might be useful, or not important. Their responses were quite different, but may have been reflected by the purposes each instructor had in using the Query Tool. Because there were only two instructors in the study, and because their individual responses are indicative of purpose, their responses were separated into two graphs, in Figure 4-3. On the graph, 2 represents important, 1 represents may be useful, and 0 represents not required. Instructor 1 (from the Transforming Teaching online course, see Table 4-1) felt contact information such as email and NSID were highly important characteristics, an opinion which is confirmed by the instructor's behavior of identifying, then contacting, individual students who were falling behind in the

material. In this case, online contact information was necessary, as it was the only means of communication between the instructor and learner. The second instructor (of the CMPT 100 regular section, see Table 4-1) felt that contact information was not necessary, likely because as a blended learning course, the instructor could communicate with the learners in person. Instructor 1's course did not have any quizzes, which is why the rating of the quiz characteristics was lower. Both instructors felt the discussion forum and participation information was important, as both courses had a participation aspect. The characteristics related to course material was also rated important by the instructors, which is shown in the types of views they created. In the Learner Statistics, Survey, and Results section, an analysis was completed on learner willingness to open these same characteristics to instructors and other peers in their course.

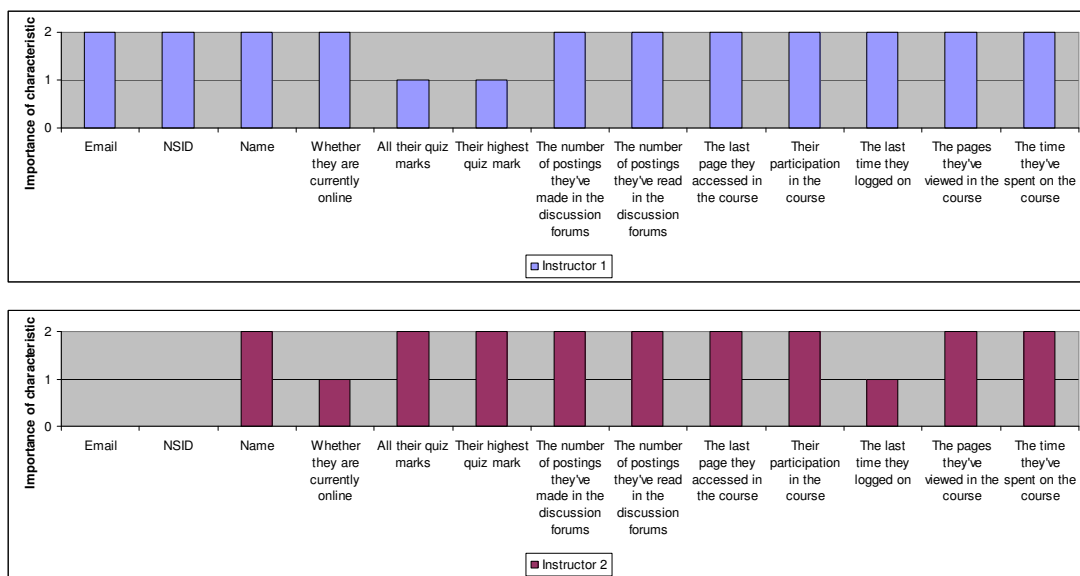


Figure 4-3: Instructor rating of importance of characteristics (a) Instructor 1 (b) Instructor 2

#### 4.2.2. Usability of the Query Tool

There are several interfaces instructors use in the Query Tool – the view creation interface, the list of view interface, and the view generation interface. The instructors were asked to evaluate the usability of each of these interfaces.

The view creation interface is the first interface instructors use, in order to set up their views. When asked if this interface was easy to use, one instructor agreed it was, while the other disagreed. The long list of characteristics on this screen appears to have caused the problem in this case. The instructor indicated that he/she was not able to find characteristics of interest using this screen, and explained it further in the area for open comments on the questionnaire: *“too many fined grained views adds a lot of clutter – it is hard to find what you want”*. As well, both instructors said the process to create a view was not straightforward. One instructor said *“all steps weren’t completely obvious – eg you had to go to a separate place to view your selection”*.

The list of views interface is the second interface instructors use, in order to display which views were available to generate. The instructors both agreed this interface was easy to use, but differed in opinion on the individual qualities of this screen that made it so. One instructor said they were able to quickly find what information was required on the screen, whereas one disagreed. This may be caused by the longer list of views available to the instructor in the second course, where the system administrator added other views the instructor did not create. One instructor was able to quickly find the views available to learners in the course, whereas the other was not. The instructor who disagreed did not choose to make views available to the learners, so may have been unfamiliar with how to determine which views were accessible.

The last interface is the view generation interface, which instructors use to produce the learner models and display the results. Both instructors agreed that this interface was also easy to use, though one instructor indicated it was still difficult to find what information was needed on this screen. This instructor’s class was very large (100+ students, divided among 3 sections), which could have contributed to this lower rating. The instructor made a suggestion which highlights this problem, and could provide a suggestion for improvements to the Query Tool: *“it was not possible to aggregate the fined grained data – e.g. show me who watched all the videos”*.

In terms of the speed of the view generation, the instructors' opinions again differed, though this may be explained by the depth of views they chose to create (see Table 4-2 and Table 4-3). The instructor who found the speed slow created views with all of the information available to them on one view, and created very general pages (for example, "Is there anyone who is falling behind in the material?", which found the name, email, time spent on the entire course, number of posts for the entire course, etc.). Since these statistics were over the entire term and for all students, the calculation took a longer time to return a result. The other instructor indicated they strongly agreed with the statement "the view is displayed quickly after I select it". This instructor created more targeted views (such as "viewed hci and web video" in Table 4-2) where only learners who viewed the videos were included in the view, and only the time spent on that material was computed by the system, thus reducing the time needed. In general, the Query Tool implementation ignored the time parameter of the active modelling function discussed in Chapter 2, in order to reduce complexity of the system. It appears in some situations, this simplification was acceptable, but in future implementations, time should be considered, instructors encouraged to use other types of views, or views limited to reduce the possibility of this occurring. Potentially, as a view is created the expected time to calculate it can be displayed to the instructor, so a better understanding of the effects of time on the view is known. Or, calculations could be done offline in preparation to generate the view, though this may lead to inconsistent data.

The tables Table 4-5, Table 4-6, and Table 4-7 show the time taken by all users able to generate the views for each course by instructors and system administrators. The times in the tables do not include an indication of context of when the views were created, which has an impact. For example, if a view is generated at the start of term before there are learners in the course, the view returns faster because there are no further computations. A view calculating the total time spent in a course returns faster at the start of term when the learners had only gone through a few pages than at the end of term when the learners had viewed most of the course. In general,

however, considering the purpose of the view, the times are quite fast. A few errant times in the tables were due to high loads on the server at the time the view was generated.

Table 4-5: Time taken to generate views in Course 1 (CMPT 100 regular section)

<b>View Purpose</b>	<b>Average</b>	<b>Minimum</b>	<b>Maximum</b>
Last log in times and time spent on course for ONLINE students	0.588s	0.078s	2.594s
Which INCLASS students are using the material and how much?	0.813s	0.218s	2.953s
Has anyone watched the lecture video?	0.164s	0.000s	1.563s
How am I doing in the course compared to other ONLINE students?	4.708s	0.360s	66.703s
What is my participation mark?	0.084s	0.000s	2.032s
Who has watched the Evolution of Computers video?	0.531s	0.406	0.703s
INCLASS general information - usage, login times	0.802s	0.578s	0.922s
who inclass saw videos module 2	4.390s	4.390s	4.390s
time spent on network videos (inclass)	2.962s	0.672s	9.953s
time spent on hci and web videos	2.657s	0.281s	7.953s
viewed web and hci video	0.692s	0.313s	1.625s

Table 4-6: Time taken to generate views in Course 2 (Transforming Teaching)

<b>View Purpose</b>	<b>Average</b>	<b>Minimum</b>	<b>Maximum</b>
General Statistics for Course	0.754s	0.407	2.953s
Usual log in times of the day	5.242s	0.157s	10.32s
Access times in the last week	1.234s	1.062s	1.454s
Contact information for students who have not yet logged on	0.166s	0.093s	0.235s
Who uses the cd?	0.111s	0.093s	0.157s
Overall time spent on the course	0.802s	0.328s	1.219s
Time spent on content for the weeks 0-4	0.536s	0.406s	0.750s
Feb. 28 Module 9	0.187s	0.093s	0.250s
test	1.578s	1.578s	1.578s
List of all students	0.218s	0.156s	0.281s
When should I log on to see someone online?	36.60s	0.718s	88.21s
Is there anyone who is falling behind in the material?	3.164s	2.750s	3.578s
Are there students actively viewing content?	0.156s	0.156s	0.156s
Are there students actively viewing content?	1.531s	0.657s	3.078s
end of course stats	0.266s	0.266s	0.266s
final course stats	0.750s	0.750s	0.750s

Table 4-7: Time taken to generate views in Course 3 (CMPT 408)

<b>View Purpose</b>	<b>Average</b>	<b>Minimum</b>	<b>Maximum</b>
Requirements for Introduction Week	0.180s	0.094s	0.266s
Requirements for Introduction Week	0.150s	0.015s	0.296s
Requirements for Privacy Week	0.187s	0.109s	0.265s
Requirements for Privacy Week	0.089s	0.000s	0.125s
How many posts have students made and started each week?	0.398s	0.000s	0.891s

In general, the instructors found aspects of the Query Tool interfaces difficult to use, particularly because for the instructor view, there are several steps (on different pages) to create views, list views, set permissions, and generate views. The process involved in going from one page to the next was not clear. The system also appeared to use inconsistent naming of items in the menu versus on the windows, and used technical terms when not all of the instructors were familiar with some of the terms. In a few situations, the time the Query Tool took to generate the view also contributed to a negative review of system usability. The difficulty in using the system could also be partially due to the administrator interface of the iHelp Courses system which was unfamiliar to the instructors. Overall, the difficulty in usability may have contributed to the low usage of the Query Tool by instructors.

#### 4.2.3. Accuracy of the Query Tool

The instructors were asked on the questionnaire if the information contained in the views was inaccurate or misleading, and further asked for explanations of what types of inaccuracies, such as inaccurate data (eg. the learner spent more time on a page than the view displayed), inaccurate views (eg. a view of participation was not really indicative of participation in the course), and inaccurate comparisons (eg. some learners performed better than their peers, but the comparisons did not convey this). The instructors both felt the information was not inaccurate or misleading.

#### 4.2.4. Effects of the Query Tool on Teaching and Learning

To analyze the effects of the Query Tool on Teaching and Learning, the instructors were first asked to rate how they felt the learners in the course were performing before and after using the system. One instructor (of the fully online course) responded he/she had no idea how the learners were performing before using the system, but after felt the learners were performing below average. So, even though the news was bad, the Query Tool was able to provide the instructor with information that could be used to make adjustments for the remainder of the course. One instructor (who instructed in a blended learning environment) responded to both questions that he/she felt the learners were performing at an average level in the course. In this case, the cues from in-class interaction likely had more effect on the instructor's impression of the learners, although potentially the Query Tool could provide further explanations as to "why" the learners were performing in the way they were (e.g. they did not participate in the class discussion because they had not all viewed the online video for the module this week).

The instructors were then asked if they had changed their teaching strategy in the course based on the results of the Query Tool, to which they both answered yes. One instructor changed the times he/she logged in to the system in order to see more learners online, and also changed some of the course content and workload to help motivate the learners to spend more time on the content. The other instructor spent more time in class reviewing the material not viewed by the learners. So, the information provided by the system changed activities of the instructors in their courses.

#### 4.2.5. Privacy and the Query Tool

The instructors both agreed that the system adequately provided learner privacy, and agreed that they received useful information from the system, even with privacy protection in place. The instructors also felt the learners received useful information from the system as well, considering the characteristics that were allowable for each



of their respective courses. Neither instructor set characteristics or filters to blocked for their learners, but at least one instructor indicated this was because he/she did not know such a feature existed. The other instructor did not choose to make views available, so there was no need to set the privacy levels.

When asked whether they believed individual learners could be identified using information in the views they created (assuming privacy protection measures were in place and learners had access to the views), the instructors said they did not think learners could be identified. However, it is interesting that when asked whether learners should be able to create views just like an instructor, one instructor agreed learners should create views, whereas the other instructor felt it was not appropriate, although the questionnaire did not explore the reasons for this response.

#### *4.3. Learner Statistics, Survey, and Results*

Only learners from one course (CMPT 100 online, see Table 4-1) were asked to complete the learner survey, as this was the only course that chose to open learner models to students. Completing the survey was optional and voluntary, and so only five students actually submitted it. Results this small can not be considered statistically significant, but the suggestions and experiences from the survey still provide unique insights into how the tool was used by actual learners. Where permission was given, usage data captured by the iHelp systems was considered in light of the learner's survey responses, using MD5-hashed user ids to ensure anonymity of the participants.

Anonymous usage data was also captured by the system across two terms of the course to determine when and how learners used the query tool. The views available to the learners were slightly different in each term, yet close enough in intention to compare the results between terms. In the first term, views were created and opened to learners for each module (see Figure 4-4 for a description of the view) and for the entire course (see Figure 4-5). Another view was created to show the learners' level of participation for the entire course (see Figure 4-6). In the second term, a general

view was created to tell the learners how they were performing in the entire course rather than each module (see Figure 4-7), along with another view to again show their participation (see Figure 4-8).

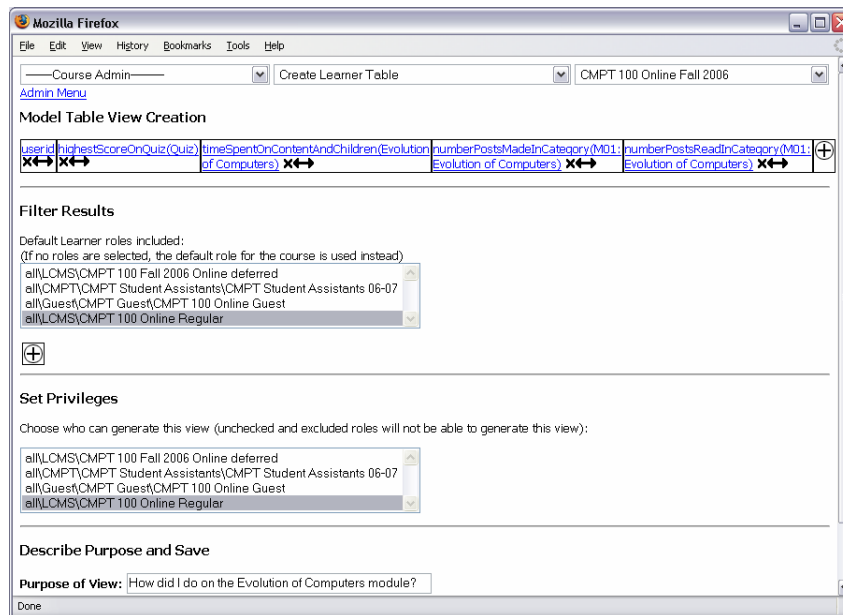


Figure 4-4: Sample view created for term 1 for each module

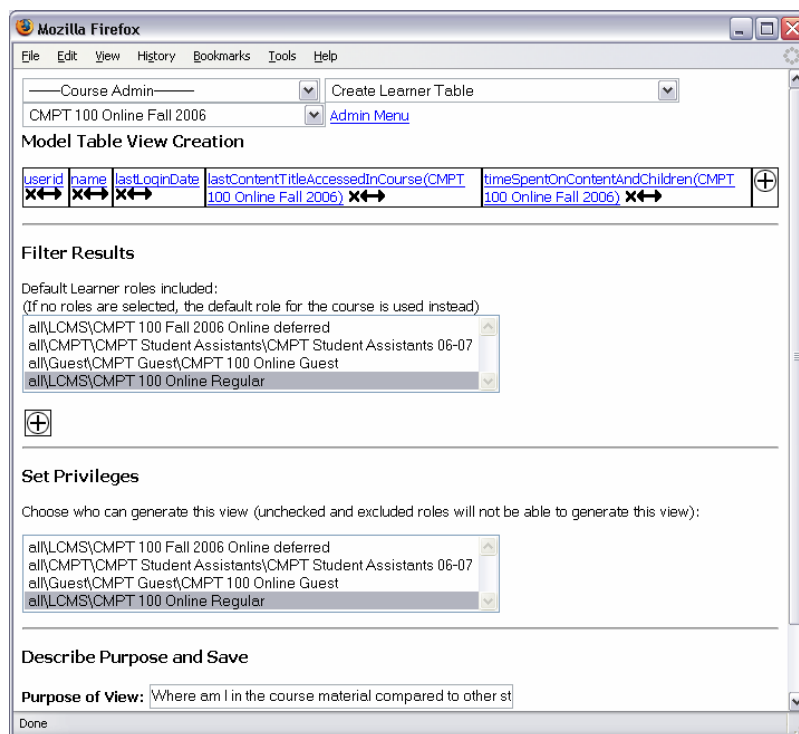


Figure 4-5: “Where am I” view created for term 1

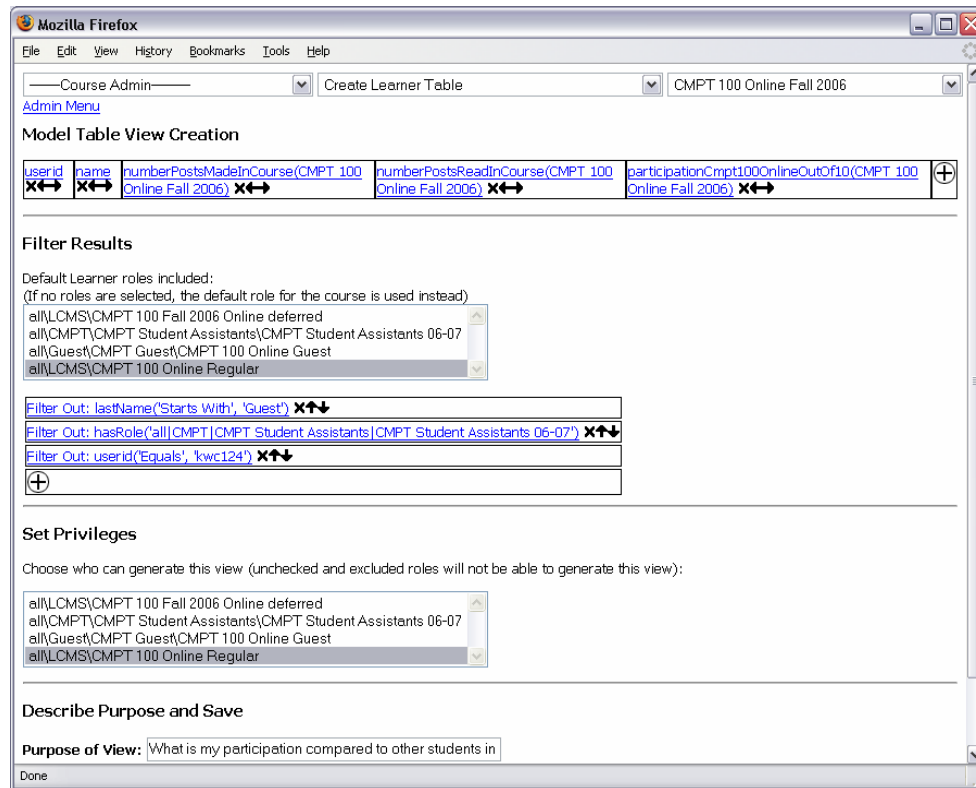


Figure 4-6: Participation view created for term 1

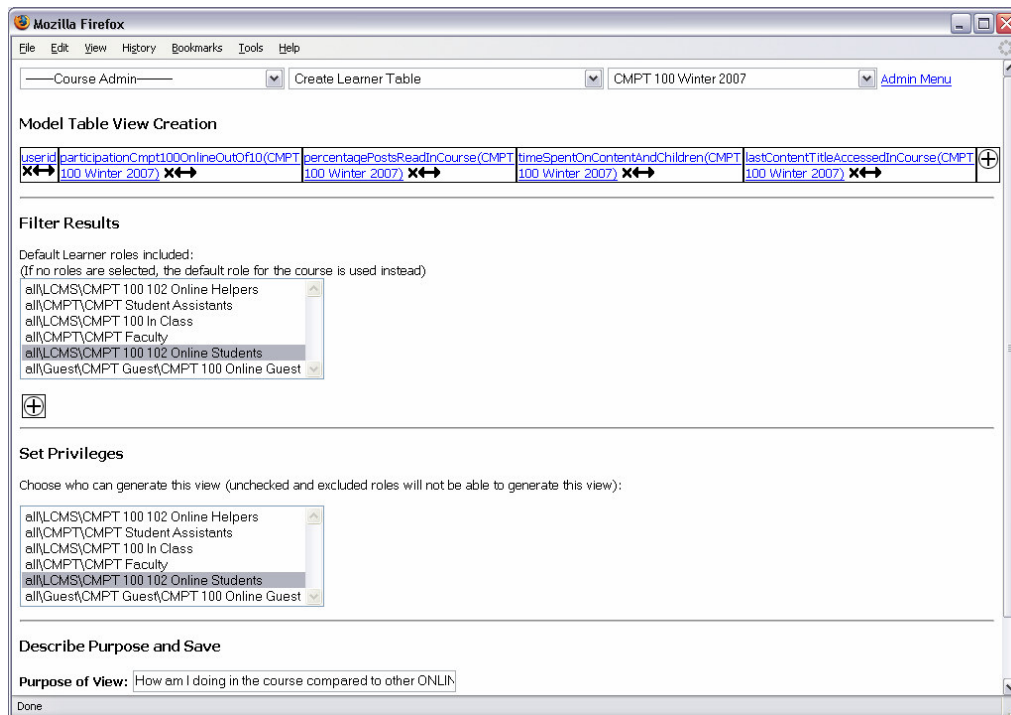


Figure 4-7: Progress view created for term 2 for entire course

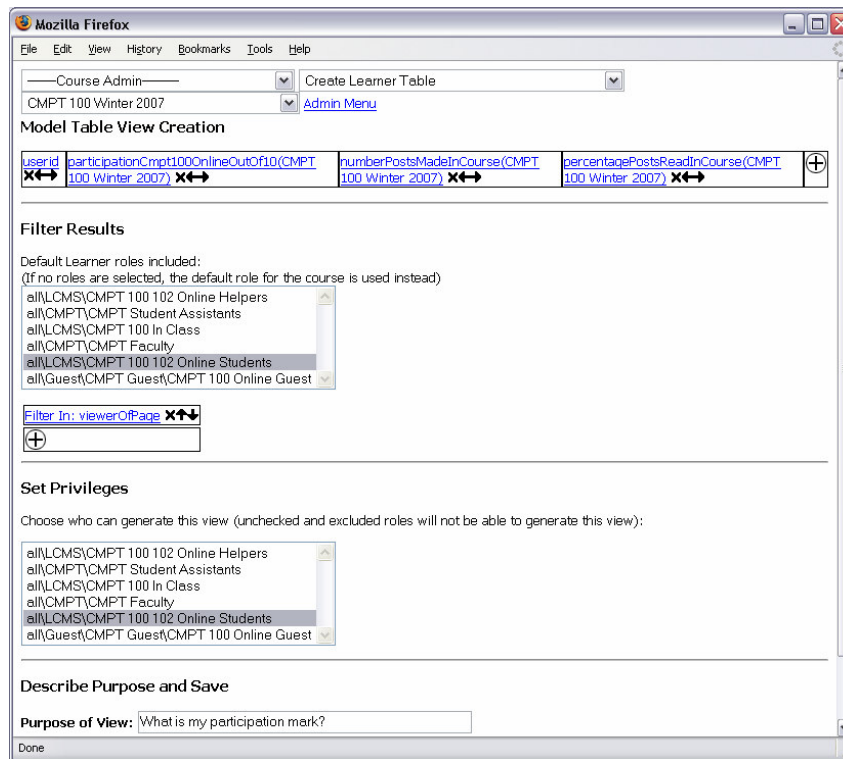


Figure 4-8: Participation view created for term 2

The students in Term 1 (26 learners in the course) used the Query Tool from the beginning of October until the end of the term. The students in Term 2 (16 learners in the course) used the Query Tool beginning in January until the end of the term. Only students in Term 2 were asked to complete the questionnaire of their experiences using the Query Tool (which to them was called the Compare Progress feature of iHelp Courses). The survey answers of the five learners and statistical data of all learners in the course were analyzed to describe the learners' experience with the Query Tool in terms of usage, usefulness, usability, accuracy, effects on learning, and effects on privacy. The findings for each of these areas are presented below and then summarized.

#### 4.3.1. Query Tool Usage and Usefulness

The learners in both terms used the Query Tool throughout the term, although there was more usage of the system at the beginning and ends of the term. Figure 4-9

shows the usage of the Query Tool throughout Term 1, showing total uses and unique learners (since some learners often used the Query Tool more than once per session). Figure 4-10 shows the usage of the Query Tool for Term 2, a course which had half as many learners as the first term, yet the system was used about equally as in Term 1, as the instructor for the course encouraged more use of the tool in the second term.

Figure 4-11 shows the breakdown of the usage of the Query Tool into four groups – learners who never used the system, used it a few times, used it often, and used it frequently. Most of the students in the courses used the system at least once or twice, and many of them used it throughout the term.

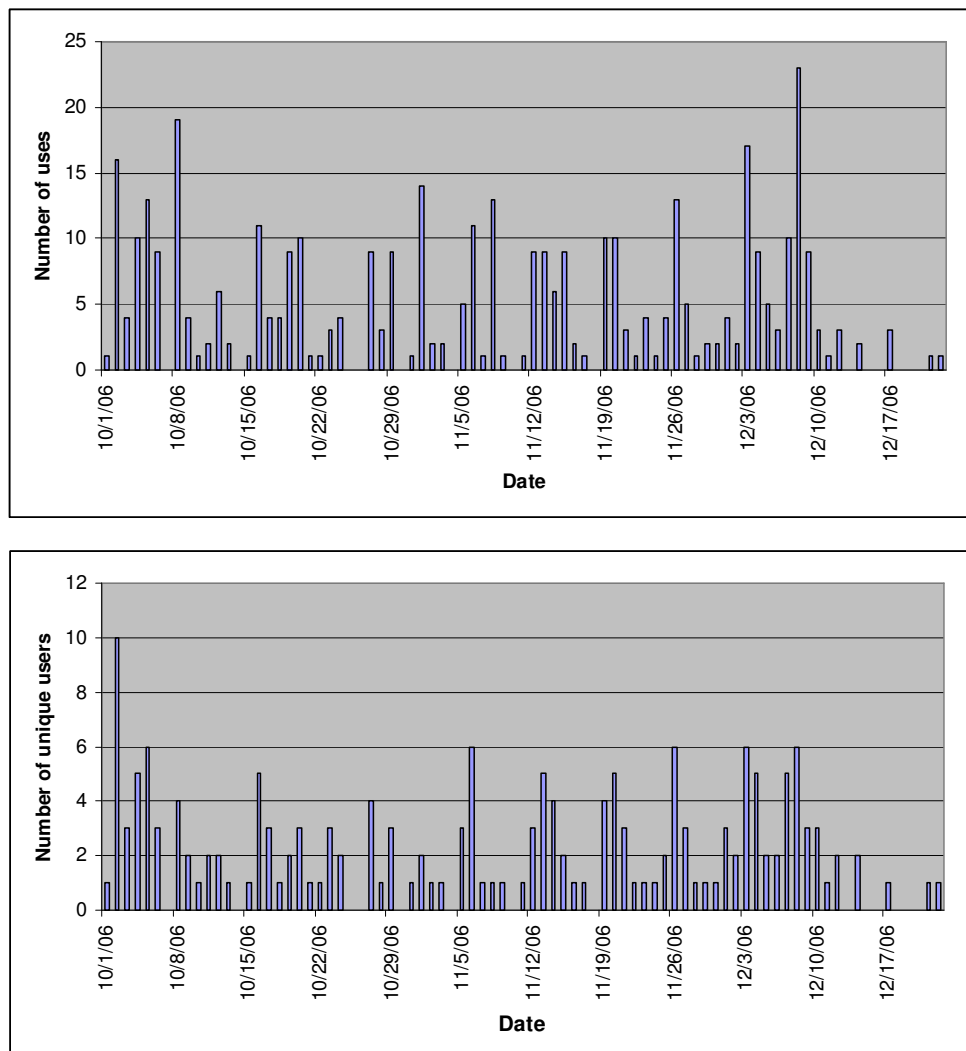


Figure 4-9: Learner usage of Query Tool in term 1 (a) Total (b) Unique Learners

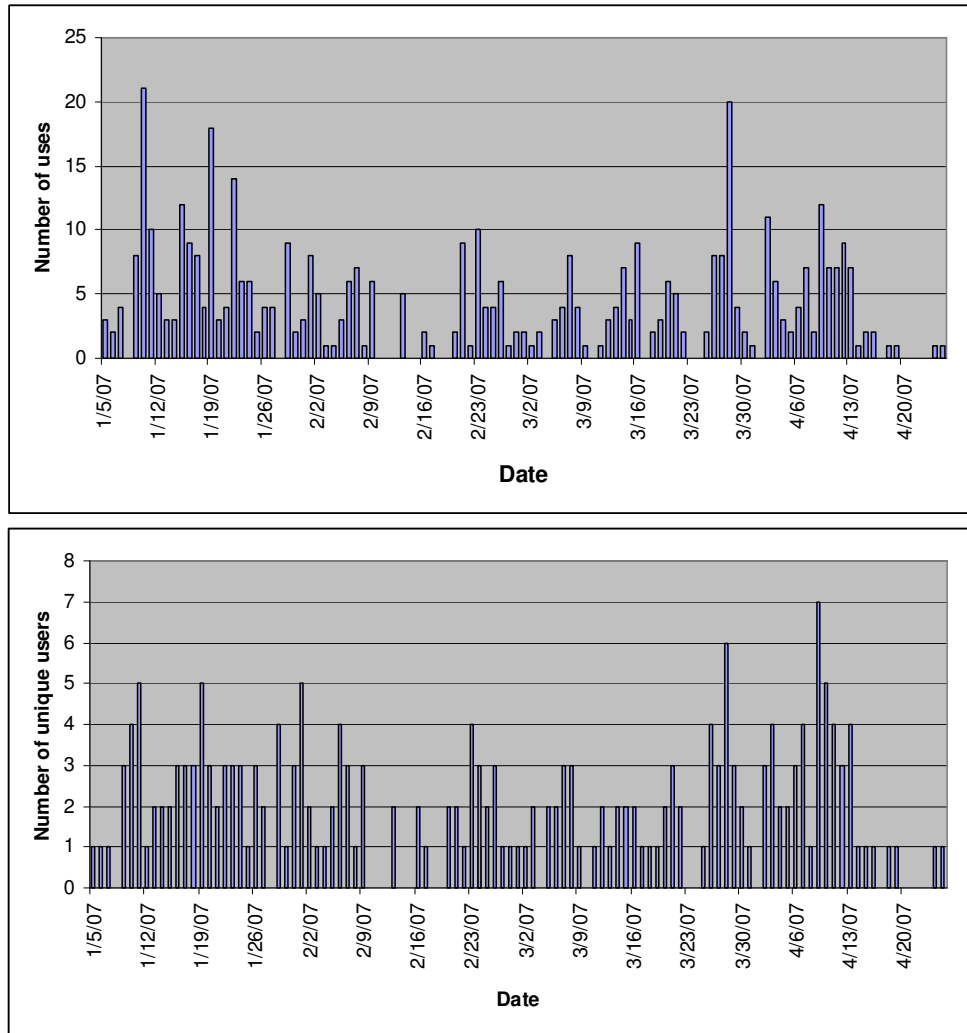


Figure 4-10: Learner usage of Query Tool in term 2 (a) Total (b) Unique Learners

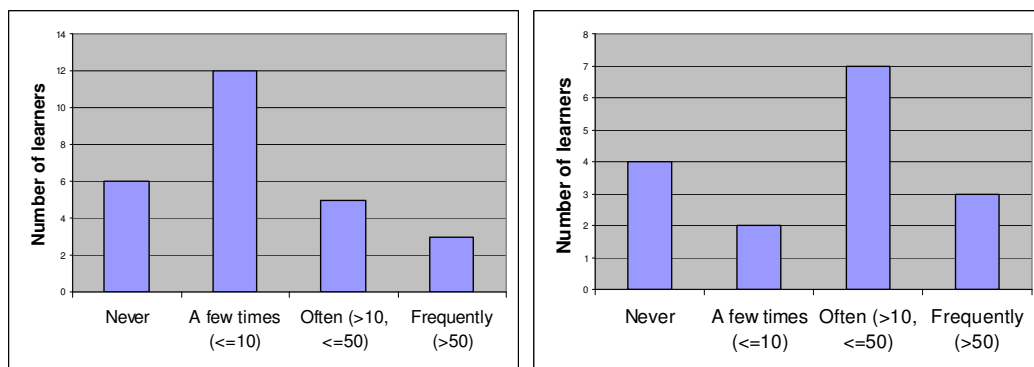


Figure 4-11: Usage of Query Tool by groups (a) in Term 1 (b) in Term 2

Considering the different types of views available to the learners (each with different end goals for the user), did they prefer one view over another? Figure 4-12 and

Figure 4-13 show the number of times each of the views were generated by learners in the courses, and show that the learners used the view “What is my participation” much more than the other views in the course. This is likely due to the allocation of participation marks for the course, but reveals an interesting capability of the system, in that learners can keep up to date with their participation mark and perhaps affect it throughout the term. In traditional courses, participation can be an “unknown” mark to learners. In any case, the learners in this course were concerned with their level of participation in the course, as the figures show.

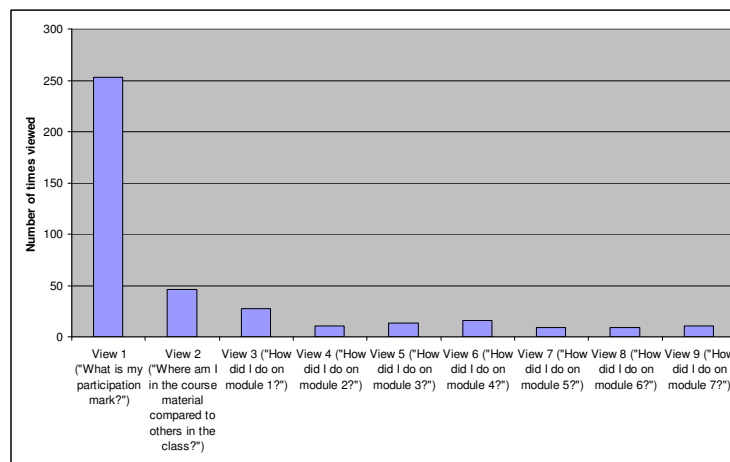


Figure 4-12: View usage in term 1

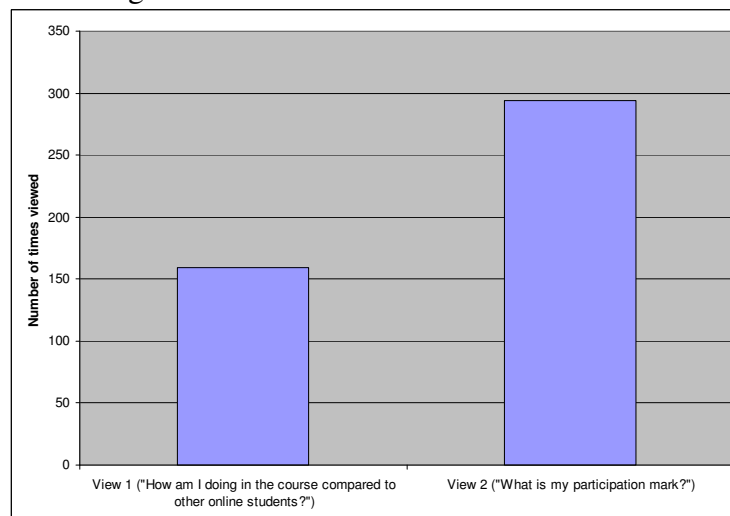


Figure 4-13: View Usage in Term 2

Figure 4-14 and Figure 4-15 show the view usage in a different way, to determine if the learners changed which view they preferred throughout the term. In Figure 4-14, View 1 (blue) is the participation view (shown in Figure 4-6), View 2 (purple) is the progress report for the entire course (shown in Figure 4-5), and “Other” (yellow) refers to the module progress reports (an example is in Figure 4-4). For Figure 4-15, View 1 (blue) is the progress report for the entire course (shown in Figure 4-7), and View 2 (purple) is the participation view (shown in Figure 4-8). The learners seem to have preferred the participation view throughout the term, though for both terms, near the end of the course several learners generated the progress reports frequently, perhaps as they studied for the final exam.

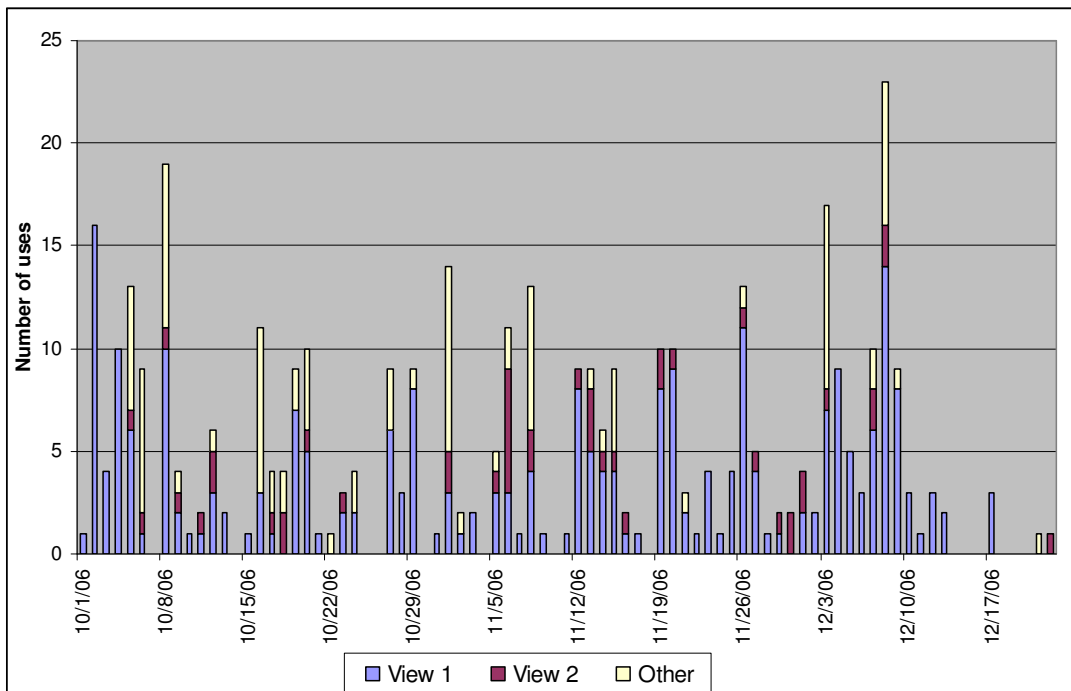


Figure 4-14: View usage timeline in term 1



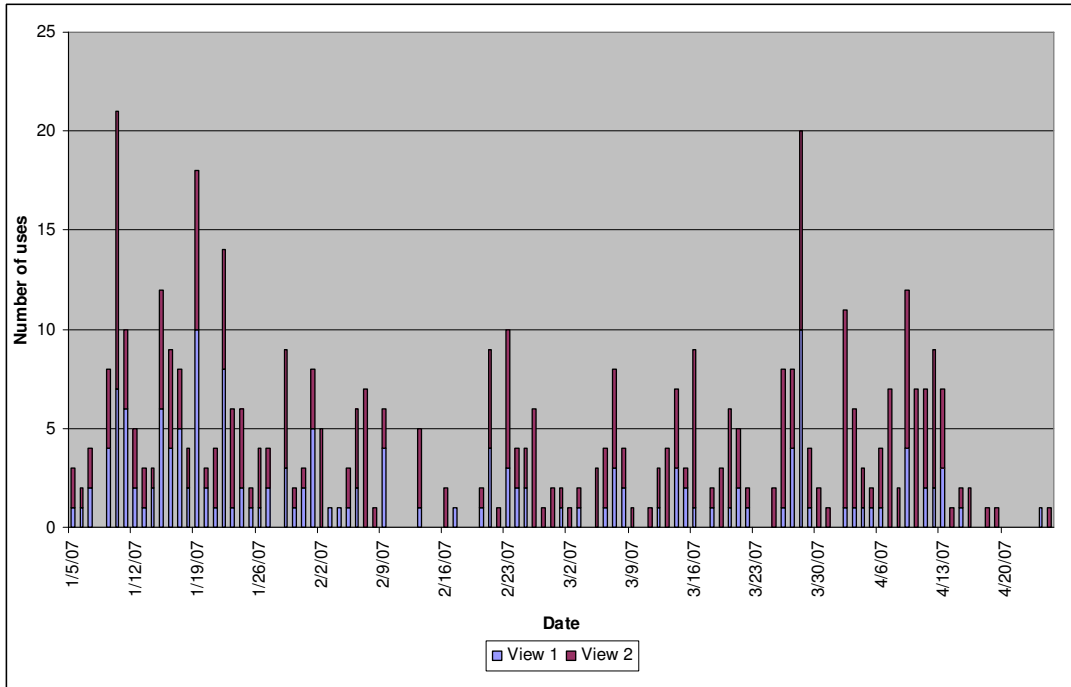


Figure 4-15: View usage timeline in term 2

While the usage statistics show that the learners used the Query Tool, the data alone does not describe whether the learners found the information presented in each of the views useful to them for their course. The questionnaires asked the learners to agree or disagree with the statement “I am able to find valuable information about my progress in my course using the Compare Progress feature”. Figure 4-16 displays the responses, which shows the learners found the Query Tool useful.

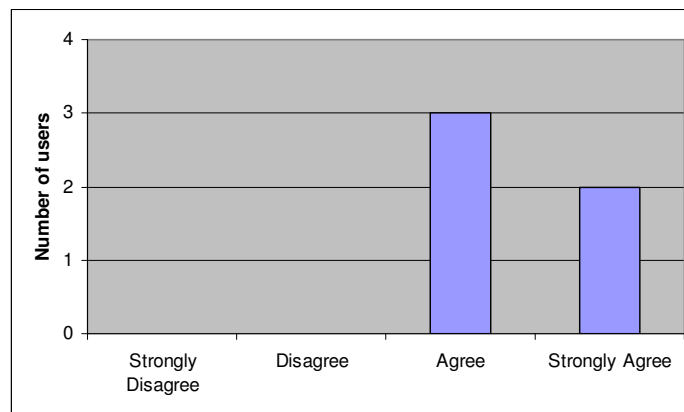


Figure 4-16: Usefulness of the Query Tool: “I am able to find valuable information about my progress in my course using the Compare Progress feature”

To examine the usefulness of the Query Tool further, the questionnaire then asked the learners to indicate their reasons for using the system, providing a list of reasons to choose as many as applied, with an open area to describe any other reasons not listed. Figure 4-17 shows their choices. One learner was of particular interest, in that he/she chose all of the choices except for the choice “to see what information someone else could find out about me”. The learners did not provide any additional reasons other than those listed in the questionnaire. The top two choices were “to make sure I am keeping up” and “to see my participation mark”, which were the descriptive names of the views indicating the purpose. Motivation to do well in the course was also a reason the learners used the tool. While the view names did not indicate a comparative purpose, several learners indicated they used the Query Tool to see if they were above average or the top learner in the course, or to see how they compared to their classmates. It is interesting that although the learners used the Query Tool to see how they were doing, they did not indicate that their goals included changing their behavior, as only one of the learners responded that one of the reasons he/she used the Query Tool was to “affect [his/her] mark”. While most of the learners claimed they did not change their behavior, studying their usage in more detail suggests otherwise (this is discussed further in the section “Effects of Query Tool on Learning”).

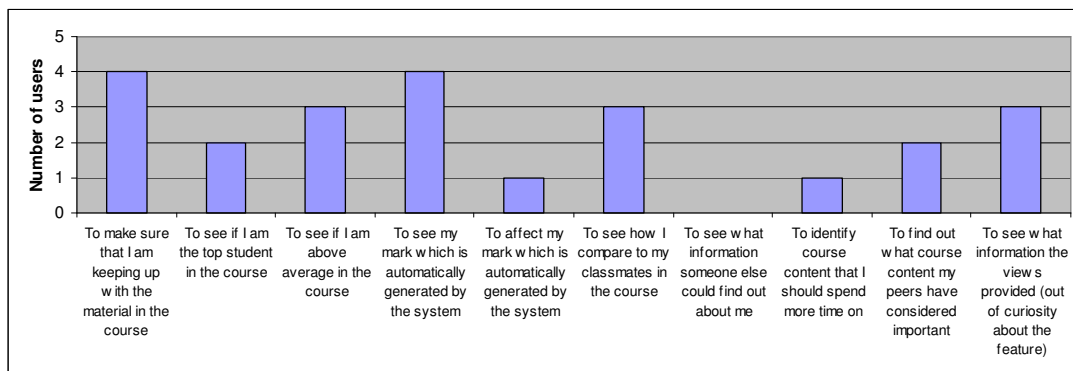


Figure 4-17: Reasons for using Query Tool

#### 4.3.2. Usability of the Query Tool

One of the goals for the Query Tool was to encourage the learners to dig deeper into the learner model by providing an easy to use interface. The questionnaire asked the learners to evaluate the Query Tool in terms of usability by answering several questions about the interface (Figure 4-18). All the users indicated the Query Tool was easy to use, and that the information presented on the view was understandable, although one user commented that the interface could be *“more aesthetically pleasing to the eye”*.

Of particular interest regarding the usability of the system was the high evaluation for the statement “The view results screen displayed quickly after I select[ed] it”. As the active modelling approach is used to generate the information right up to date when the view was called, the high evaluation shows that the method used was typically fast for the learners for the views they were generating. Looking at the usage logs, the average, minimum, and maximum times taken to generate the views for the two terms of interest are shown in Table 4-8 and Table 4-9. The graphs shown in Figure 4-19 and Figure 4-20 show the time taken to generate the views over the term by considering only learner uses of the views. While there are a few longer times (caused by higher loads on the server at the time), most of the response times at the start of the term were quite low and expectedly increased near the end of term as data in the system grew. Considering the computations and permission filtering performed on the system backend, this shows the approach has generally worked well for the views opened to these learners, although some consideration for scale should be considered in later development, particularly if more complex views are opened.

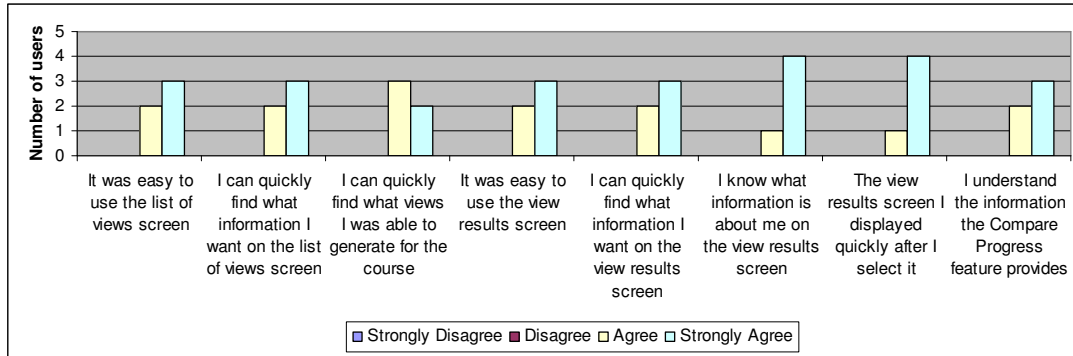


Figure 4-18: Usability of the Query Tool

Table 4-8: Time taken to generate views in Term 1 considering only student usage

	Average	Minimum	Maximum
<b>View 1</b>	1.038s	0.609s	2.641s
<b>View 2</b>	3.883s	0.969s	9.829s
<b>Other Views</b>	0.951s	0.453s	2.313s

Table 4-9: Time taken to generate views in Term 2 considering only student usage

	Average	Minimum	Maximum
<b>View 1</b>	4.948s	0.406s	66.703s
<b>View 2</b>	0.086s	0.000s	2.032s

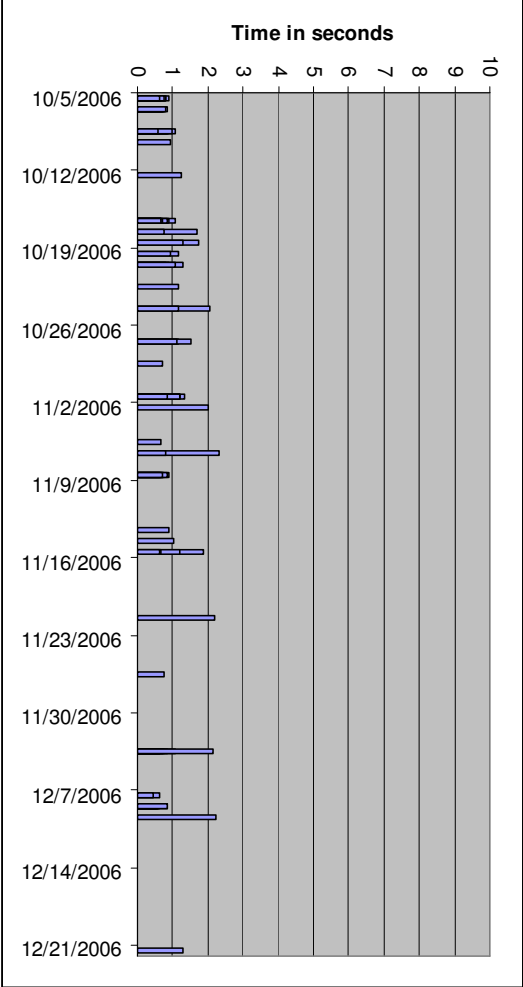
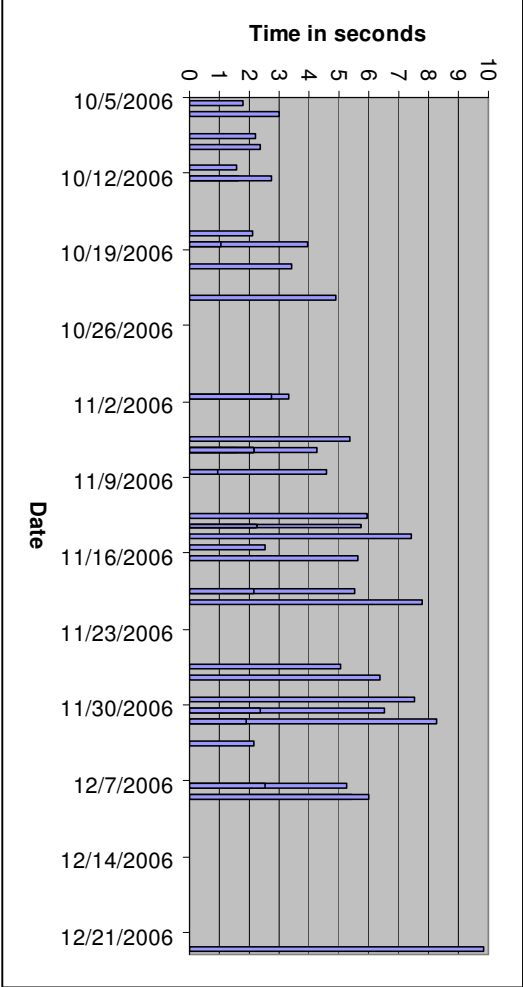
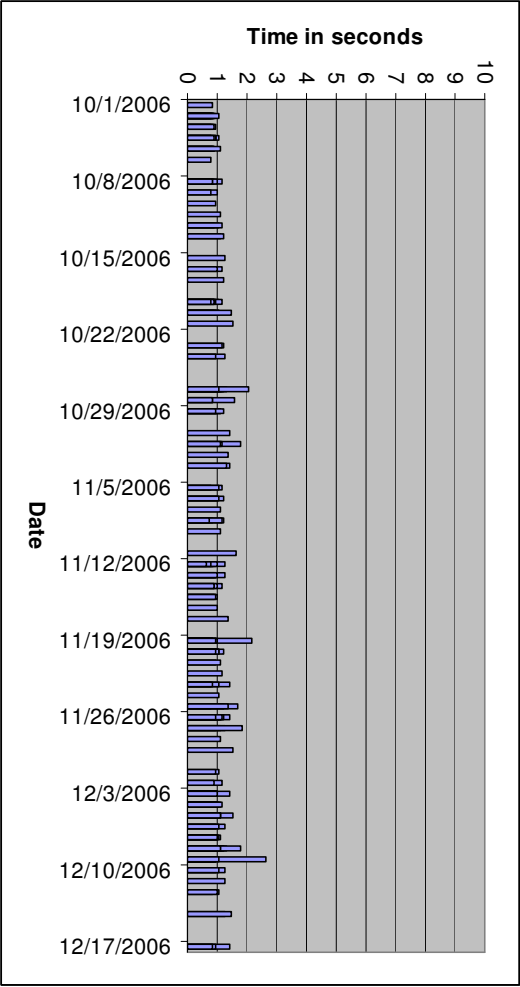


Figure 4-19: Time taken (in seconds) to generate views in term 1

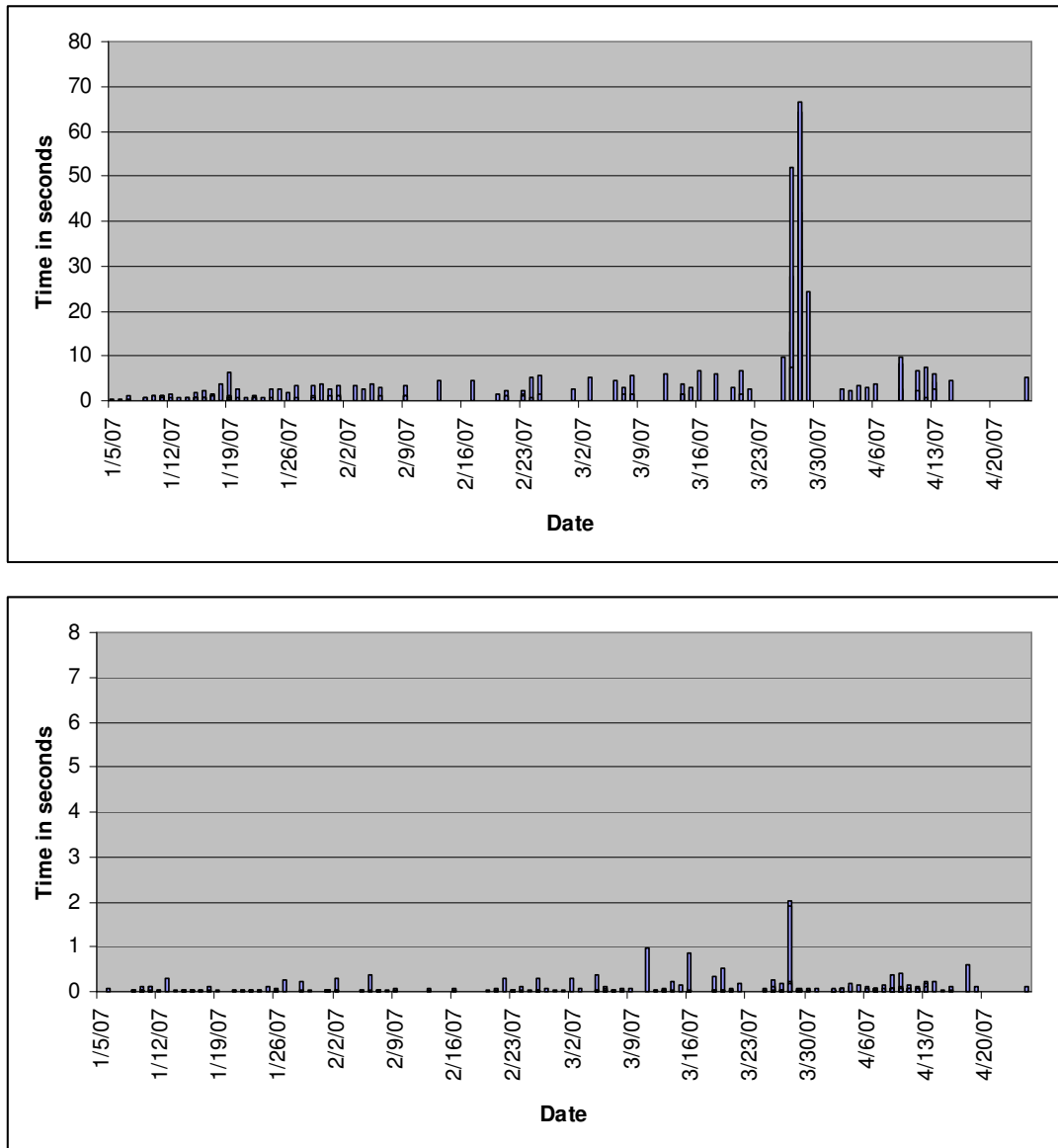


Figure 4-20: Time taken (in seconds) to generate views in term 2

#### 4.3.3. Accuracy of the Query Tool

To judge the accuracy of the views, the learners were asked if the information on the views was inaccurate or misleading. Their responses varied on this question – most felt the information was not misleading, although one learner thought the information was inaccurate (Figure 4-21). The learner who agreed with the statement explained further: *“time spent on the course material...what if we just left the webpage open for a long period of time?”* One of the limitations of web

browsers is the difficulty in determining when a user has closed a window. The Query Tool does not include in any computations time spent on pages longer than a few hours, as it is assumed the learner left the page and the system was unable to capture this action. The learner may have noticed this limitation while using the system.

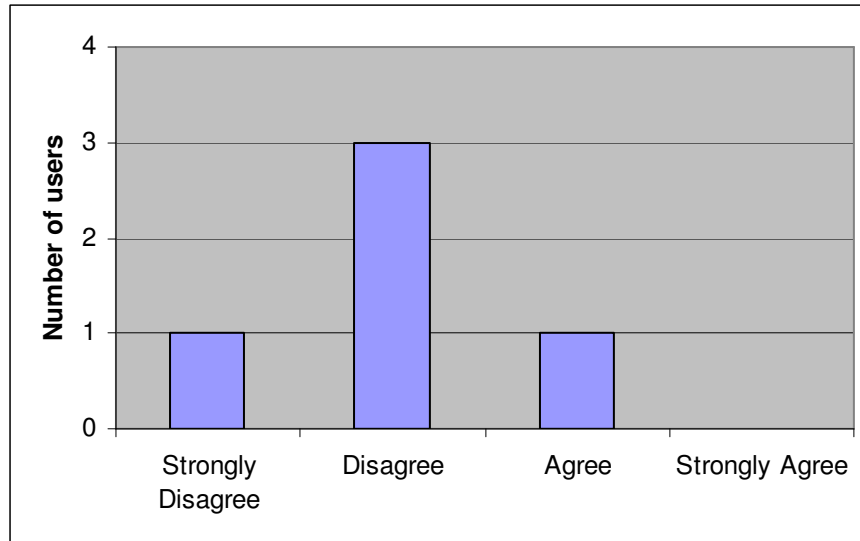


Figure 4-21: Accuracy of the Query Tool: “The information contained in the views is inaccurate or misleading”

One of the learners wrote that one of the positive aspects of the Query Tool is that “[it]shows your current mark, which is updated and accurate”. This same learner often re-ran the views in a single session after viewing content or posting messages in the discussion forums. The up-to-date calculations thus increase the perception of the Query Tool’s accuracy, as students were able to directly observe the changes their actions invoked on the information contained in the view.

One learner wrote that one of the negative aspects of the Query Tool is that “any glitches where marks are not calculated right and one goes over all the info repeatedly to [find] out where that last ~20% of material not covered is hiding!” This comment related to an initial miscalculation of the participation for the course by including discussion categories not available to the learners in the online section

of the course. The problem was identified by learners who, according to the discussion forums, had read all of the messages posted in the category while the Query Tool reported they had only read about 80% of the messages. This problem was quickly fixed, and learners reported after that the updated percentages were correct.

Overall, however, the system was perceived as accurate by the learners.

#### 4.3.4. Effects of the Query Tool on Learning

The learners were asked to describe the extent to which the Query Tool affected their behavior in the course (Figure 4-22). Most learners indicated the information in the views had little to no effect, although one learner indicated the information had a significant effect. In clarifying what they changed after using the Query Tool's views, the learners described reviewing course materials while studying. One learner said "*[I used the system to ensure] that all of the material was covered - I may have overlooked some aspect of it before so it acts as a reminder*". Another learner voiced a similar objective: "*I reread certain areas that I felt I was weak in.*" The results from this question were surprising – it was expected that the learners would make significant changes in their behavior after determining how they were performing, as this is an objective of open modelling for reflection. Perhaps one reason for this result is that learners who completed the survey typically had higher participation and activity in the course, and so the comparisons shown on the views were sufficient to meet or surpass their goals. Or, the particular kinds of reflective activities encouraged by the Query Tool were not sufficiently strong enough to have any effect.



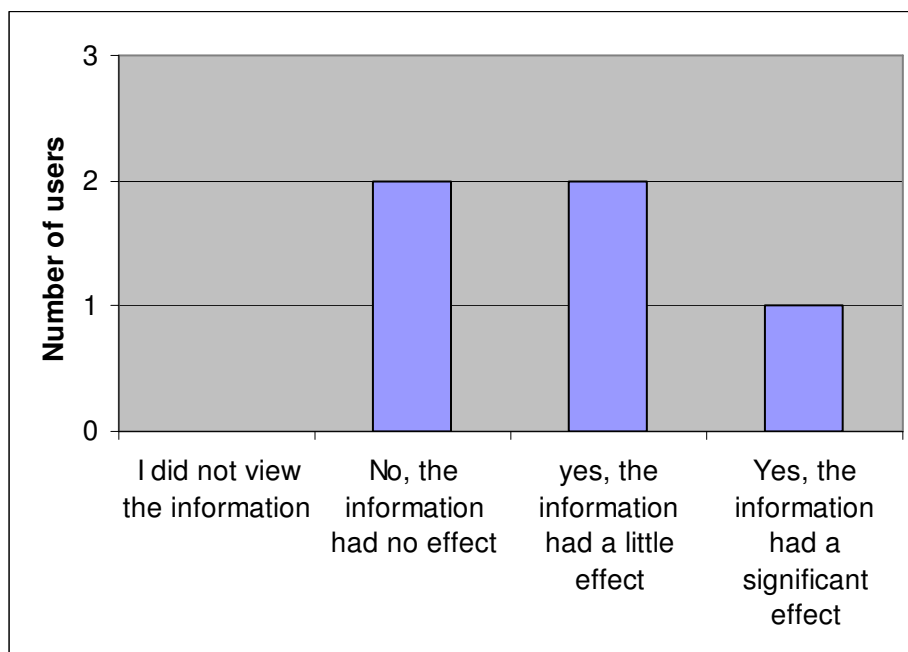


Figure 4-22: Learner perception of Query Tool effect on Learning: “Information contained in the views changed my behavior in my course”

While the learners claimed to have not been influenced by the information in the views to change their behavior, some of the usage patterns actually suggest the information did have some effect. One of the learners who answered that the information had no effect would often log on to the iHelp Courses system, generate the view “how am I doing in the course”, then log off soon after. This behavior was repeated several times, particularly near the end of the term. When the view revealed that his/her participation was very high, the learner quickly logged off. When the view revealed that his/her participation had dropped, the learner spent some time in the discussion forums before logging off soon after. A similar pattern of behavior was observed in several other learners. This suggests the learners used the Query Tool to do “just enough of”, but not more than, what was expected of them regarding participation in the course. This may be a positive or negative consequence of the Query Tool – negative in that it may curb learner activity beyond what is required, and positive in that the learners know enough about their progress to establish and reach goals for the course. There has been some research into the reasons learners “game” a tutoring system (purposefully use qualities of the system

only to score better rather than fully learn the material) [BAKER2005]. This research has suggested having performance goals (as the student appears to have here in achieving a high participation) does not lead to a negative impact of learning a topic, and in some cases gaming the system even helps. In any case, this consequence of the Query Tool needs additional study.

To determine if the Query Tool had any further effects on the social environment for the course, two timelines were created – one to compare the number of postings made in the discussion forums by learners in the course to the number of Query Tool uses, and one to compare the number of postings read by learners in the course to the Query Tool uses (Figure 4-23 and Figure 4-24). While it is difficult to distinguish whether the Query Tool affected the class discussion or whether the results indicate learners used all components of their online course on the same day, there are many spikes in the discussion forums that match spikes in Query Tool uses. External factors outside of the study may have had an impact here – the online course has weekly course deadlines and weekly class discussion topics. Students working harder to meet these deadlines may have been making sure they had reached the participation requirements for the course.

Participation in the course consisted of weekly discussion topics, but the learners were aware there was not a “deadline” for each discussion topic. Rather, they could contribute to the discussion throughout the term, or even get participation credit if they posted all their messages at the end of the term for all discussions. Most of the learners contributed to the class discussions throughout the term, as Figure 4-23 indicates. The learners in the course were aware that the instructor also generated the views frequently to monitor participation in the course. This raises the question whether the Query Tool provided a form of social pressure on the learners, since they knew their progress through the term was being observed by their instructor and peers. One of the other courses using the Query Tool (CMPT 408, see Table 4-1) also had a participation component for the course and a flexible deadline for online discussion messages. However, in that course the learners largely opted to wait until

the end of the course to post messages. This offers an interesting speculation – if the CMPT 408 course had weekly reports through the Query Tool and students could see their participation compared to their classmates in a quantitative visualization, would they too choose to contribute weekly to the discussion?

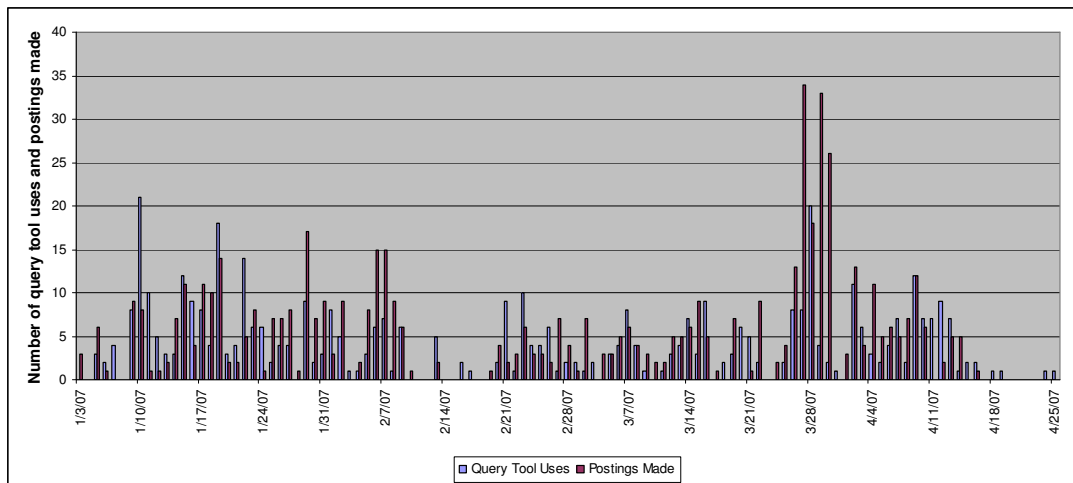


Figure 4-23: Query Tool usage and postings made in iHelp Discussions in term 2

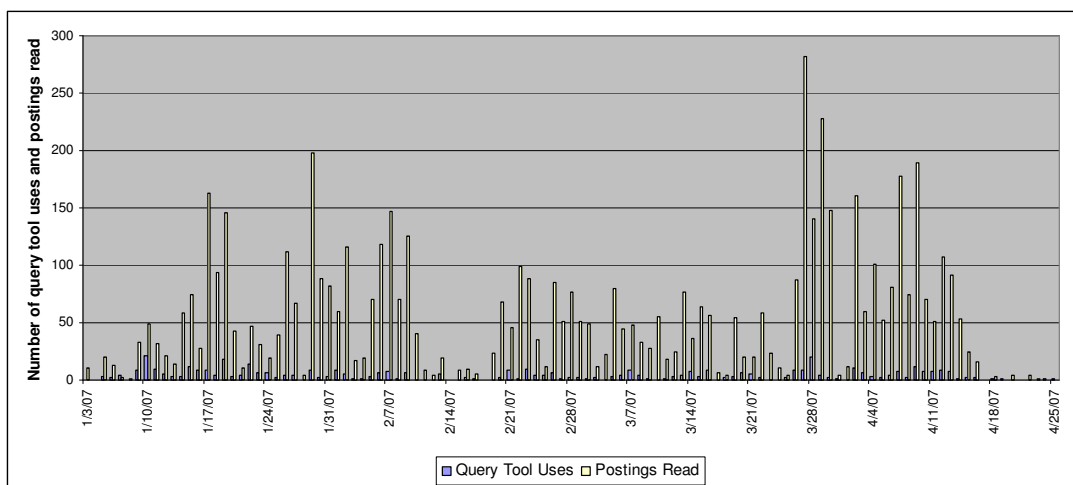


Figure 4-24: Query Tool usage and postings read in iHelp Discussions in term 2

Learner motivation was also influenced by using the Query Tool. The learners were asked to rate how they viewed their performance (below average, average, above average, or no idea) before using the Query Tool, then after using the Query Tool, using the same scale (Figure 4-25). Most learners considered their progress as average before using the views, yet this increased to above average after. All

learners who completed the survey indicated an increase in performance. These results may again be affected by the group of users who completed the questionnaire (learners at the top of the participation and activity in the course), but do show how the Query Tool can be used as a positive encouragement to learners who may not see their progress as adequate. By being able to compare themselves to others in the course, these learners were able to see a more appropriate picture, and this may have further influenced their activity.

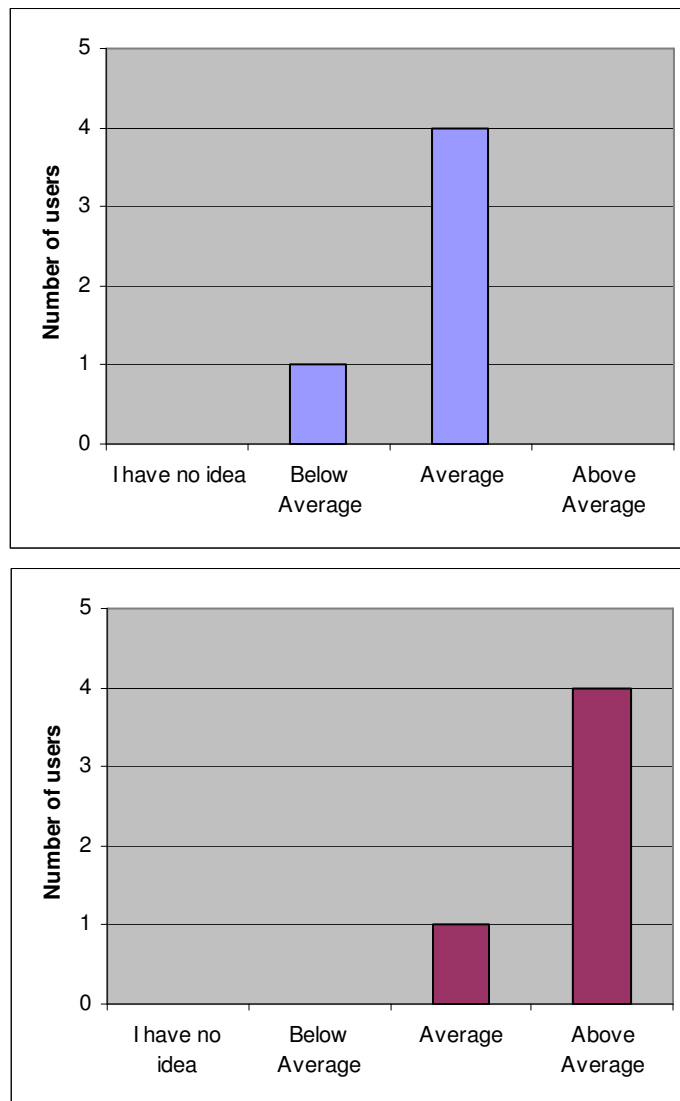


Figure 4-25: Learner motivation and perception of performance (a) Before using the Query Tool (b) After using the Query Tool

One learner wrote that one of the most positive aspects of the Query Tool was “*you can compare to other students and for me, it was motivation to do better - competition is sad, but it's a great motivator.*” Another learner listed as both a positive and negative aspect of the Query Tool that “*[it] may scare people into thinking that they haven't done enough work/spent enough time*”. The learners frequently used the tool for motivation of increased performance. A comment from the questionnaire sums this up: “*When I discovered this feature, I thought that it was such a great idea; I was able to view my compared marks and participation to others in the course - it motivated me to do better in the course itself, and in turn, I learned more than I would have without it.*”

#### 4.3.5. Privacy and the Query Tool

As seen in the previous section, opening information about learners to their peers in a course enables comparison and thus a sense of how well one is doing in a course. However, opening this information to others also opens up issues of privacy, as this is sensitive personal information (as discussed in Chapter 3). The Query Tool enables privacy protection by blocking characteristics and filters which could potentially be used to identify particular individuals. When the learners were asked about the privacy protection provided by the Query Tool, their reaction was positive, all agreeing that privacy was protected adequately (Figure 4-26). Most of the learners indicated that privacy on the web was a concern for them, so a high rating of the Query Tool in terms of privacy protection is important. The learners were also asked if the system still provided useful information to them, even though some of the information was blocked due to privacy concerns. They responded that they agreed the system remained useful despite the privacy protection.

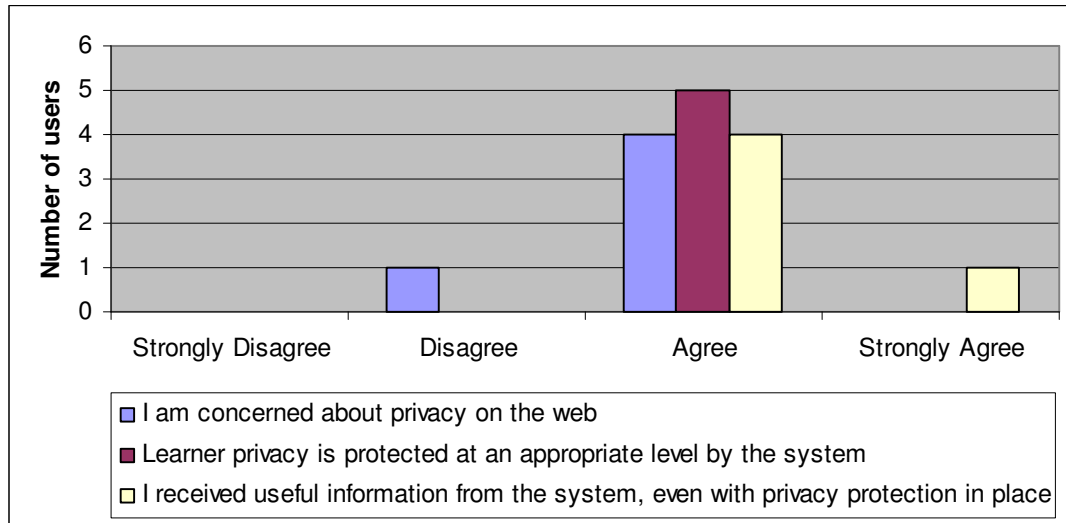


Figure 4-26: Privacy protection provided by the Query Tool

Learners were asked if they felt it was possible to identify another learner using the views available to them in the course, and whether they had actually identified someone. The results for this were not clear – on one hand, most of the learners felt it was not possible to identify another learner, but when they were asked whether they identified someone, most indicated that they may have, but were not sure. One learner indicated that he/she thought it was possible to uniquely identify someone, and thought of way to do this, but had not actually done it for their course. Still, the learner indicated on their questionnaire that privacy was adequately protected by the Query Tool. Unfortunately, the questionnaire did not provide an area for the learners to comment on either of these situations.

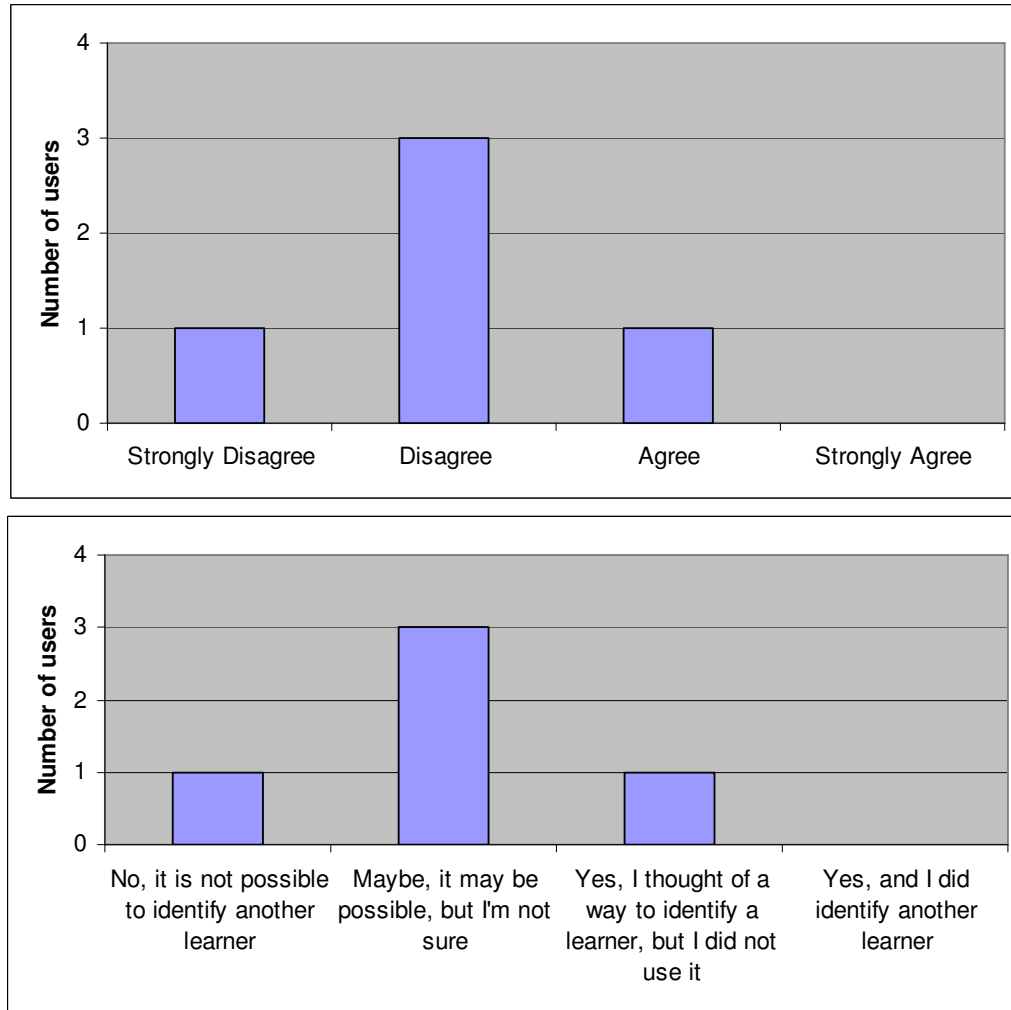


Figure 4-27: Identifying another learner using the Query Tool (a) "I believe it may be possible to specifically identify another learner using information on the views I had access to in my course" (b) "I was able to specifically identify another learner using information provided by the Compare Progress feature"

Certain characteristics are of a more sensitive nature than others, although this may be user-specific. For example, some individuals may not want their name to be known to all of their peers. Others may only want instructors to be able to see their quiz marks. Yet others are willing to reveal all information to their classmates. The questionnaire asked the learners to comment on what characteristics they would like to see about others to effectively compare themselves to their peers, then asked to whom they would make these characteristics available if requested. Figure 4-28 and Figure 4-29 show the responses. Figure 4-28 shows the average calculation of importance as supplied by all of the learners, where 0 means the learners together

felt the characteristic was not important to be able to compare themselves, and 2 means the learners felt the characteristic was vital to compare their progress. Most of the information was rated as “nice to have but not important”, including identifying information such as NSID and name. Two of the characteristics considered important (participation and time spent in the course) were available to the students using the views in their course, but the third (quiz marks) were not made available to them. It was interesting to note that although the students found that the participation was very important, the characteristics included in the calculation for participation (the number of postings made and read in the discussion forums) were not considered as important, meaning students may only need aggregate or computed forms of the information rather than the entire result. One student put it this way: *“I think that it is nice to know where you sit individually and in comparison to the top, average and lowest... but I don't think it is really necessary to show all of that information to everyone! (not that it is a bad thing - just not as useful!)”*

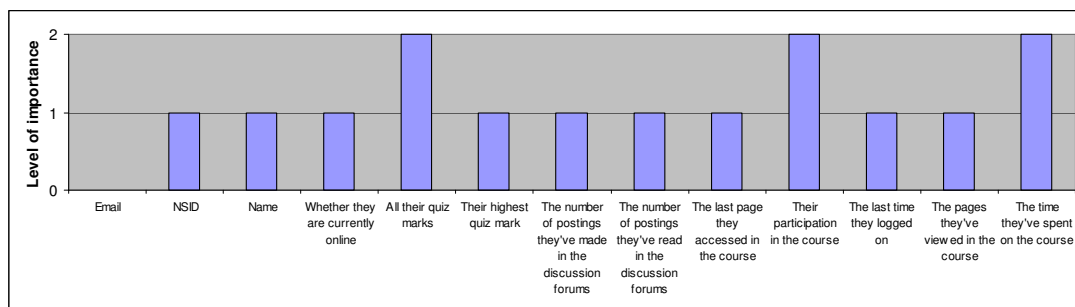


Figure 4-28: Importance of characteristics (0=Not important, 2=Important)

The questionnaire also asked the learners to consider the groups with whom they would feel comfortable sharing certain characteristics. The characteristics included their name, NSID (a unique user identification required to access computer services at the university), discussion activities, and course activities. The groups were their classmates, their friends, and their instructor. The high level of openness shown by the learners who completed the survey was surprising (Figure 4-29), as it was assumed that learners would choose to block more information than the results



indicated, particularly from their peers. These learners had no problems opening up most of the characteristics on the questionnaire to their peers. Most of the learners thought they would be comfortable revealing their name, although fewer wanted their contact information (email) to be displayed to their peers. The learners did not feel that discussion activities (showing the number of messages they posted or read) posed any threat to privacy concerns. Course activities varied. Showing whether they were online was not considered a concern, but several students indicated that they would not want their marks on the quizzes shown to their peers. There was also not a large difference in the number of discussion- and course-related characteristics that the learners felt they would show to their friends but not to their peers, which may indicate that the learners felt there was not a significant difference between these groups when observing personal educational information related to their course. However, there was a difference in the contact information that the learners would make available to their friends but not their peers, as they would allow their email to be displayed to their friends. The learners felt comfortable displaying all of the characteristics to the instructor of their course.

The general openness of these learners to display most of the information in the system may be reflective of the type of learners who completed the survey. These learners all were active in the course, completing most of the course content and participating in the class discussion. Therefore, the information displayed about them on the views was quite positive, which may influence their decision about whether to open it to others. If the information was negative, this may have changed their opinion of what information they would be comfortable showing to others.

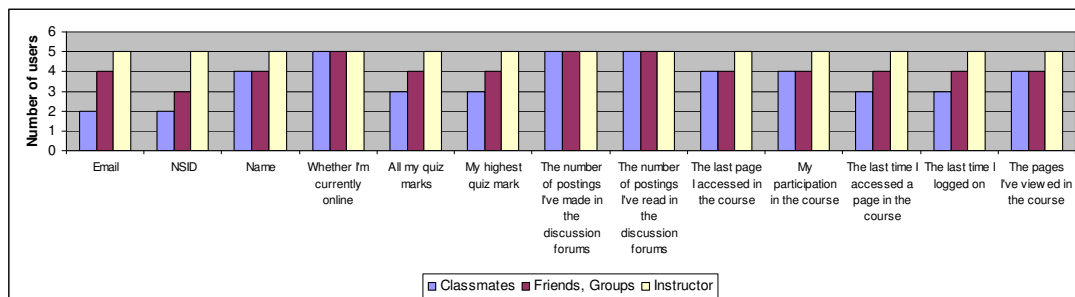


Figure 4-29: Level of openness to other learners, friends, and instructors

#### *4.4. Limitations of Study*

The study had a few limitations which affected analysis of the responses. The number of participants in the study, in particular the number of instructors, gave only partial results of the types of views users created and generated in the system. As well, during analysis of the questionnaires, a few questions required additional clarification from instructors, but this was not enabled by the questionnaire. Interviews or further follow up would have provided more sources of information in these cases.

#### *4.5. Summary and Conclusions from Study*

Both the instructors and learners involved in the study felt the Query Tool was useful to them for their course, and used the system in different ways. Instructors used the system to observe how learners were interacting with the system and changed their interactions with the learners appropriately. Learners used the system as a motivation to perform better in the course by comparing themselves to others. Interesting patterns of behavior were observed when analyzing the usage of the system, which may be construed as either positive or negative, depending on the situation and goals of opening the learner model. The system was used significantly by learners when views were made available, though instructors did not use the Query Tool as much as was expected. This appeared to have been caused partially by the interface and workflow of the system, and partially by the limitations of the data available to the system. The approach taken in development of the system (active computation) was also effective in that results were up-to-the-minute, but some compromise was made in some situations in terms of the time taken to generate some of the larger views. Privacy also appeared to be adequately provided by the system (in the opinion of both instructors and learners), although some external influences may have been identified as enabling the identification of individual learners, and warrants further exploration of the privacy effects of active open learner models. Throughout the study, it was observed that context had a high impact of what information to include in an open learner model. The high impact of

context was theorized at the beginning of this research. Overall, the study showed that the Query Tool is useful, particularly for fully online courses.

## 5. SUMMARY AND CONCLUSION

### 5.1. *Summary*

Determining whether the research goals have been achieved can be determined in light of the questions posed in the intention statement from Chapter 1:

- To build a query tool to be used by learners and instructors to explore their learner models based on questions the learners have about social and cognitive aspects of the models.

Chapter 3 discussed the functionality of the Query Tool using real queries (also called views) taken from several online and blended learning courses that were created by instructors. The instructors created learner models to answer such questions as “Who is falling behind in the course”, “How well are the learners performing”, and even material-related queries like “Is the material appropriate”. Learners used instructor-created queries to answer questions related to activity in their courses: “How am I performing in the course compared to my peers”, and “What is my participation in the course?” There are other questions (or purposes) the learner may have when opening the model and the system offers the advantage of being extendible to capture other kinds of information, although this is constrained by the kinds of data captured by the online delivery system. The Query Tool provides a large degree of flexibility, and provides the ability for instructors in different courses (whether online or blended learning) with different requirements to still gain useful information from the system.

Chapter 4 discussed the effects of the Query Tool as evaluated by instructors in several courses and learners from one online course, and it was found that the use of the Query Tool was appreciated as it gave instructors and learners valuable insights into how their respective courses were progressing. The system was used frequently by learners to observe their progress through the term, although the instructors only used the system occasionally, partially blaming the interfaces of the Query Tool and the delivery system for this inactivity.

- To use active modelling to compute the learner models and develop appropriate visualizations.

The Query Tool uses the purely active approach in computing the learner model. That is, the model is computed in real time when the user requests it, with only targeted information included in the model depending on the context (the purpose for creating the learner and viewer of the model). The active approach offers benefits to open learner models but introduces problems, namely how to establish this needed context and purpose. The Query Tool used the instructor's opinion to establish what to include in the model by enabling him/her to include characteristics and learners in the learner model (or view) as appropriate to the purpose.

Only one visualization was implemented in the system (a table view) for simplicity, and it was found to be limited and largely visually unappealing to instructors and learners, though both groups found this type of view was still useful to fill their purposes. However, other types of visualizations are possible using the same approach.

- Develop a balance of system usefulness and privacy protection as the learner model is opened to instructors and peers as well as the learner for comparative purposes.

One of the goals of the Query Tool was to make information about a learner's peers available to other learners in the course, so the viewer could adequately compare himself/herself to others in the course to establish how he/she is performing. This is an activity difficult for learners in an online course that does not have the usual physical cues and environment that face-to-face courses provide. However, in order to do this, privacy must be protected so as to enable learners to feel safe in their learning environment. In particular, information that allows identification of a particular learner should be protected. The Query Tool does allow a level of privacy protection by allowing instructors to limit certain characteristics in the learner model from being accessed by particular groups of users. This does still offer problems (as discussed in Chapter 3) as one or two kinds of information have the potential to be misused to identify learners. However, such misuse is difficult to monitor or even predict. Instructors and learners who used the Query Tool found that the system adequately protected privacy, although it appears a few had doubts about the abilities of the system regarding this aspect. Even with privacy protection in place, the users found that they still received useful information from the system.

- To test the Query Tool and learner models in terms of performance, robustness, and appropriateness in real-world learning environments.

The real world often adds complexities, in learning environments especially, since learners are dynamic, always changing. Chapter 4 discussed the use of the Query Tool and evaluation of the system using real users in online and blended learning courses in order to evaluate the performance of the learner models created. Overall, the system performed well in terms of speed, usually returning the learner models in a few seconds when requested. A few factors limited the performance, namely the number of students in the course, the number of learners in the view, and the complexity of the learner model

being generated. Even so, the users (both learners and instructors) found the model accurately represented how learners were performing in their course, but did express the desire to be able to obtain more information from the system. The Query Tool also appropriately reached pedagogical goals of the instructors who used the system – namely to determine how their students were performing and how effective online materials were for the course.

## *5.2 Future Directions*

As discussed in Chapter 4, a pilot study was completed with a few online courses, and evaluated only the responses from a few learners in these courses. Most of the learners who completed the questionnaire were top students in their course, so various other types of learners were under-represented. Although the questionnaire had few participants, the results of the questionnaire did provide insights into how learners in a real learning environment used the Query Tool. The effects of the Query Tool should be analyzed in a larger setting with additional types of learners to determine the effects of opening the learner models to learners in the course.

One of the findings from the study discussed in Chapter 4 was that the interface used by the instructors was difficult to use, as there were many steps in the process that were unclear. This likely contributed to the lack of use of the system by instructors. Another future direction of the Query Tool is to create a more suitable interface that is easier to use for instructors, and provide adequate training for the system so instructors are able to create the views they are interested in. However, the system allows for very complex models to be created, and there is a large amount of information that can be included in the learner model. It will be difficult to balance a more user-friendly interface with adequate functionality and extendibility the system could provide. A possible solution to this problem is to include standard views for each course automatically which can be refined by instructors, or offer a wizard to import views from other courses. Another possible solution is to add a little intelligence to the system by recognizing common views from other courses or previous course offerings to recommend views to instructors or potentially even

learners. A community query pool where members contribute their own queries to the group while maintaining private query areas may also be useful to cut down on the “information overload” experienced by some of the instructors from Chapter 4.

The table view visualization chosen for the Query Tool appeared to be limiting to the instructors, and unappealing visually to learners. Other visualizations are also possible using the same underlying programming methodology (as Chapter 3 discussed), although even these could be potentially expanded as well. The questionnaire provided to instructors and learners asked users’ impressions of small visualization features that may make the system better. One such feature that was described was adding aggregate data to the columns of the views to show average, maximum, and minimum values. Another feature included adding styles or colors to the columns and rows to indicate additional information such as the top learner, learners to watch, etc. These and other more complex information visualization techniques would be highly valuable to indicate problem areas quickly to instructors.

Generating a view by using the Compare Progress feature of iHelp Courses need not be the only way a view could be generated either. In the iHelp Courses side menu, visualization bars can be added beneath content titles to indicate different characteristics of other learners that the learner may access indicate how well he/she is performing with respect to others in the course [BROOKS2006a]. In Figure 5-1, the visualization indicates how much time this user has spent on the course material (on the light colored bar), and how much time the average learner in the course has spent on the material (the dark colored bar). This side menu is always visible to the learner, and updates with each interaction in the course, so the learner’s progress report is always up to date. Information from the Query Tool could potentially be used in the same manner (although currently this is not a feature of the Query Tool).



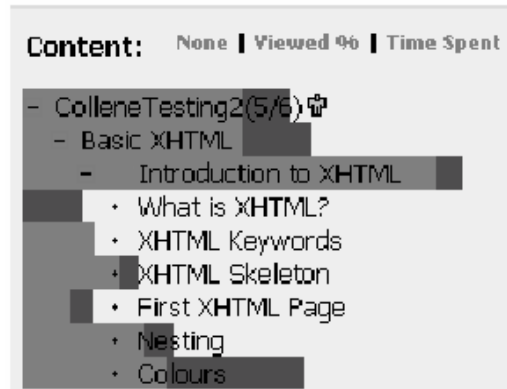


Figure 5-1: Social awareness in iHelp Courses

The Query Tool could also include additional traditional characteristics in the model, such as knowledge of topics or misconceptions as deduced characteristics computed from raw data. These were omitted from this work in order to focus on the modelling process and implementation rather than researching complex ontologies, reasoning algorithms, or data mining techniques already being explored by others. While the raw data used by the system does provide a large amount of information that can be interpreted as needed by instructors, a better indication of student performance can be determined by using reasoning algorithms [BROOKS2004] and data mining techniques to provide information to the learner model. Data mining could also potentially be used for auto discovery of views and learners as patterns are recognized in the data as learners interact with a learning environment, a method similar to the ecological approach [MCCALLA2004] [BROOKS2006b].

One assumption made at the beginning of the implementation of the Query Tool was that the effect of the time and resource parameters of the active modelling formula would be ignored in order to explore the other (arguably more important) parameters: learners (subject and object) and purposes. As Chapter 4 showed, some of the queries took a long time to return results because the purely active approach (minus the resource parameters of the formula) was taken to compute the models in real-time. By using a hybrid approach to active modelling, previously computed fragments of existing models could instead be used for these models by calculating some of the information offline. A hybrid approach would be absolutely necessary

as more complex data is added to the models as was discussed above, as such calculations would likely take a longer time. Perhaps entire models could be generated offline if the instructor is willing to compromise the up-to-the-minute benefit of the current system. This of course introduces complications into the modelling process, as some pieces of the model may be accurate while others are expired. As well, many useless computations may be done creating these partial models that are never needed. While a solution has initially been explored to use the same underlying technology and programming to solve this problem, it warrants further exploration.

Another possible direction of this research is to include an interactive mechanism similar to the work by Dimitrova [DIMITROVA2003] where users are able to input their own opinions as additional data into the learner model. To the Query Tool, this is essentially another data source, so the information can be added just as the other characteristics are added to the system. This does introduce potential problems into the learner model (in light of the warning given by Waern [WAERN2004] about the negative impact of allowing learners to adjust profiles), but also allows instructors and peers to gain feedback directly from the person being modeled in the system. This undoubtedly would provide explanations for what the viewer is seeing in the model, and give the learner being modeled the opportunity to justify actions in the system or results in the learner model.

Finally, the implications of privacy in the Query Tool must also be explored as the learner models continue to be opened to additional users. The Query Tool used a lower-level privacy protection mechanism by allowing the instructor to choose which characteristics and filters were to be allowed or blocked by the system. The study in Chapter 4 indicated that the users felt this was an adequate protection of privacy considering the views they had access to. Learners also indicated they would feel comfortable revealing information on a per-characteristic basis to other individuals in their learning community, a process that does give the learner the greatest level of control over their personal information. However, others in the

open modelling community have explored higher-level privacy protection by allowing entire models to be allowed/blocked from both groups of learners and individuals [BULL2005]. In this case, learners themselves specified who was permitted access to their model. In the case of an active open learner model, looking at privacy at the level of entire views may be more appropriate, considering the privacy discussion of Chapter 3. Because the active open learner modelling approach has the natural tendency to create many smaller but targeted models, users would be able to generate the models then decide on a per-model basis whether to allow others to view the learner models (i.e. whether to release the learner models to groups like the entire class, or individual learners like the teaching assistant or instructor). This might offer a potential solution to the discussion in Chapter 3 of blocking deduced characteristics from a seemingly harmless model of other information, as learners are aware of what the learner model is saying about them. The best approach might be to take the middle ground between the lower level and higher level privacy protection by offering a mix of both. This is even further complicated when allowing learners create their own views instead of using views created by an instructor, as learners may worry about what others might be able to model using the system. However, allowing learners to create their own views would provide many reflective benefits and encourage the learner to explore the learner models further. It would also provide a better way to determine reflective goals of the learner, and thus increase the number of useful purposes in the Query Tool.

Considering the range of directions the Query Tool may take, privacy protection is a complex problem in this research.

### *5.3. Conclusion*

This thesis has explored the idea that information in an open learner model depends on the context in which it is generated. This allows a learner model to contain exactly the right information to suit the needs of the learner at a particular time. Exploring this idea using online and blended courses has led to some understanding

of both the benefits of the approach and areas for further exploration. Benefits include producing a model generally very quickly as needed and offering privacy protection for the learners but still offering useful information when opening the learner model to other individuals in their community. Areas for future research include extended methods of privacy protection by enabling users to choose to whom the learner models are opened, enlarging the range of data available to users in the system, exploring the generation of other kinds of modelling information, providing better and more varied visualizations, and looking into the possibility of learners creating their own views. Overall, the users of the system in a real e-learning environment appreciated the use of the Query Tool as it gave them a glimpse into the progress of the community. The thesis shows that the active approach is a promising direction for future development of interesting approaches to open learner modelling.

## REFERENCES

- [AHN2007] Ahn, J., Brusilovsky, P., Grady, J., He, D., and Syn, S. Open User Profiles for Adaptive News Systems: Help or Harm? *Proceedings from the 16<sup>th</sup> International World Wide Web Conference*, Banff, Alberta, Canada, May 8-12 2007, pages 11-20.
- [BAKER2005] Baker, R.S., Roll, I., Corbett, A.T., Koedinger, K.R. Do Performance Goals Lead Students to Game the System? In Looi, C., McCalla, G., Bredweg, B. and Breuker, J., editors, *Artificial Intelligence in Education*, Volume 125, pages 57-64.
- [BAKER2000] Baker, M. The roles of Artificial Intelligence and Education research: A prospective view. *The International Journal of Artificial Intelligence in Education*, Volume 11, pages 221-241.
- [BRETZKE2003] Bretzke, H., and Vassileva, J. Motivating Cooperation in Peer to Peer Networks, *Proceedings from User Modelling (UM) 2003*, Johnstown, PA, pages 218-227.
- [BRNA1999] Brna, P., and Self, J. Negotiated Collaborative Assessment Through Collaborative Student Modeling. In Morales, R., editor, *Workshop on Open, Interactive, and Other Overt Approaches to Learner Modeling (Proceedings from 9th International Conference, AIED 1999)*, pages 35-42.
- [BROOKS2006a] Brooks, C., Hansen, C., and Greer, J. Social Awareness in the iHelp Courses Learning Content Management System. From the Workshop on Social Navigation and Community-Based Adaptation Technologies at the 2006 International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH2006), June 20-23, 2006, Dublin, Ireland.
- [BROOKS2006b] Brooks, C., and McCalla, G. Towards flexible learning object metadata. *The International Journal of Continuing Engineering Education and Lifelong Learning*, Volume 16, Nos 1/2, pages 50-63.
- [BROOKS2005] Brooks, C., Kettel, L., and Hansen, C. Building a Learning Object Content Management System. In the Proceedings from the World Conference on E-Learning in Corporate, Healthcare, and Higher Education (E-Learn 2005), October 24-28, 2005, Vancouver, Canada.
- [BROOKS2004] Brooks, C., Winter, M., Greer, J., McCalla, G. The Massive User Modelling System (MUMS). *Proceedings of the 7th International Conference on*

*Intelligent Tutoring Systems (ITS2004)*, Aug. 30 – Sep. 4, 2004, Maceio, Brazil, pages 635-645.

[BRUSILOVSKY1999] Adaptive and Intelligent Technologies for Web-based Education. *Kunstliche Intelligenz*, 4 (1999), pages 19-25. Also available online at <http://www2.sis.pitt.edu/~peterb/papers/KI-review.html>. Last accessed January 14, 2005.

[BULL2007] Bull, S., and Kay, J. The SMILI Open Learner Modelling Framework. In Brna, P., editor, *International Journal of Artificial Intelligence in Education Special Issue: Open Learner Models: Research Questions*, Volume 17(2), 2007, pages 89-120.

[BULL2005] Bull, S., Abu-Issa, A.S., Ghag, H., and Lloyd, T. Some Unusual Open Learner Models. In Looi, C., McCalla, G., Bredweg, B. and Breuker, J., editors, *Artificial Intelligence in Education*, Volume 125, 2005, pages 104-111.

[BULL2005b] Bull, S., Mangat, M., Mabbott, A., Abu Issa, A., and Marsh, J. Reactions to Inspectable Learner Models: Seven Year Olds to University Students. In Kay, J., Lum, A., and Zapata-Rivera, D., editors, *Learner Modelling for Reflection to Support Learner Control, Metacognition, and Improved Communications*, AIED Workshop 11, Amsterdam, July 18-22, 2005, pages 1-10.

[BULL2003] Bull, S., McEvoy, A.T., and Reid, E. Learner Models to Promote Reflection in Combined Desktop PC/Mobile Intelligent Learning Environments. *In the Proceedings of the Workshop on Learner Modelling for Reflection, 11<sup>th</sup> International Conference on Artificial Intelligence in Education*, AIED2003, pages 199-208.

[BULL2002a] Bull, S., and McCalla, G. Modelling Cognitive Style in a Peer Help Network. *Instructional Science* 30 (6), November 2002, pages 497—528.

[BULL2002b] Bull, S., and Nghiem, T. Helping Learners to Understand Themselves with a Learner Model Open to Students, Peers, and Instructors. In Brna, P., and Dimitrova, V., *Proceedings of Workshop on Individual and Group Modeling Methods that Help Learners Understand Themselves, International Conference on Intelligent Tutoring Systems 2002*, page 5-13. Retrieved from the World Wide Web on February 5, 2003: <http://www.eee.bham.ac.uk/bull/papers/ITS02-WS.htm>.

[BULL2001] Bull, S., Greer, J., McCalla, G., Kettel, L., and Bowes, J. User Modeling in I-Help: What, Why, When and How. In Bauer, M., Gmytrasiwicz, P., and Vassileva, J., editors, *User Modeling 2001 (Proceedings from 8th International Conference, UM 2001)*, pages 117-126.

[BULL1997] Bull, S. See Yourself Write: A Simple Student Model to Make Students Think. In Jameson, A., Paris, C., and Tasso, C., editors, *User Modeling (Proceedings from 6th International Conference, UM 1997)*, pages 315-326.

[BULL1995a] Bull, S., Brna, P., and Pain, H. Extending the Scope of the Student Model. In *User Modelling and User-Adapted Interaction* Issue 5, 1995, pages 45-65.

[BULL1995b] Bull, S., and Pain, H. 'Did I say what I think I said and do you agree with me?': Inspecting and Questioning the Student Model. In Greer, J. (ed), *Proceedings of World Conference on Artificial Intelligence and Education (ACCE)*. Charlottesville, VA, 1995, pages 501-508.

[CHEN2007] Chen, Z., Chou, C., Deng, Y., Chan, T. Active Open Learner Models as Animal Companions: Motivating Children to Learn through Interacting with My-Pet and Our-Pet. In Brna, P., editor, *International Journal of Artificial Intelligence in Education Special Issue: Open Learner Models: Research Questions*, Volume 17(2), 2007, pages 145-167.

[CHENG2005] Cheng, R., and Vassileva, J. Adaptive Reward Mechanism for Sustainable Online Learning Community. In Looi, C., McCalla, G., Bredweg, B. and Breuker, J., editors, *Artificial Intelligence in Education*, Volume 125, pages 152-159.

[COLLINS1997] Collins, J., Greer, J., Kumar, V., McCalla, G., Meagher, P., and Tkatch, R. Inspectable User Models for Just-In-Time Workplace Training. In Jameson, A., Paris, C., and Tasso, C., editors, *User Modeling (Proceedings from 6th International Conference, UM 1997)*, pages 327-337.

[DIMITROVA2003] Dimitrova, V. Maintaining Diagnostic Interactions that Promote Learners' Reflection. In *the Proceedings of the Workshop on Learner Modelling for Reflection, 11<sup>th</sup> International Conference on Artificial Intelligence in Education, AIED2003*, pages 228-237.

[DIMITROVA2003b] Dimitrova, V. StyLE-OLM: Interactive Open Modelling. In *User Modelling and User-Adapted Interaction* Issue 13, 2003, pages 35-78.

[DIMITROVA2002] Dimitrova, V. Self, J., and Brna, P. The Design and Implementation of a Graphical Communication Medium for Interactive Open Learner Modelling. In S. Cerri, G. Gouaderes & F. Paraguacu (eds.) *Intelligent Tutoring Systems Proceedings of ITS2002*, Springer-Verlag, pages 432-441.

[DIMITROVA2001] Dimitrova, V., Self, J., and Brna, P. Applying Interactive Learner Models to Learning Technical Terminology. In Bauer, M., Bmytrasiwicz, P., and Vassileva, J., editors, *User Modeling 2001 (Proceedings from 8<sup>th</sup> International Conference, UM 2001)*, pages 148-157.

[DIMITROVA2000] Dimivtrova, V., Self, J., and Brna, P. Involving the Learner in Diagnosis—Potentials and Problems. Tutorial at Web Information Technologies: Research, Education and Commerce, Montpellier, France, May 2000.

- [DIMITROVA1999a] Dimitrova, V., Self, J., and Brna, P. STyLE-OLM--An Interactive Tool in a Terminology Learning Environment. In Morales, R., editor, *Workshop on Open, Interactive, and Other Overt Approaches to Learner Modeling (Proceedings from 9th International Conference, AIED 1999)*, pages 25-34.
- [DIMITROVA1999b] Dimitrova, V., Self, J., and Brna, P. The Interactive Maintenance of Open Learner Models. *Proceedings from the 9<sup>th</sup> International Conference of Artificial Intelligence in Education*, Le Mans, France, July 19-23, 1999.
- [FINK2000] Fink, J., and Kobsa, A. A Review and Analysis of Commercial User Modeling Servers for Personalization on the World Wide Web. In Kobsa, A., editor, *User Modeling and User-Adapted Interaction: An International Journal, Issue 10 (2000)*, pages 209-249.
- [FORD2000] Ford, N. Cognitive Styles and Virtual Environments. In *Journal of the American Society for Information Science, Volume 51, Issue 6* (April 2000), pages 543-557.
- [GARRET2005] Garret, J. Ajax: A New Approach to Web Applications. February 18, 2005. Available online at <http://www.adaptivepath.com/publications/essays/archives/000385.php> . Last accessed July 14, 2007.
- [GREER2001] Greer, J., McCalla, G., Vassileva, J., Deters, R., Bull, S., and Kettel, L. Lessons Learned in Deploying a Multi-Agent Learning Support System: The I-Help Experience. *Proceedings of AIED 2001*, San Antonio, pages 410-421.
- [GREER1999] Greer, J. E., Zapata-Rivera, J., Ong-Scutchings, C., and Cooke, J. Visualization of Bayesian Learner Models. In Morales, R., editor, *Workshop on Open, Interactive and Other Overt Approaches to Learner Modeling (Proceedings from the Ninth International Conference, AIED 1999)*, pages 9-13.
- [HOLDEN1999] Holden, S., and Kay, J. The Scrutable User Model and Beyond. In Morales, R., editor, *Workshop on Open, Interactive, and Other Overt Approaches to Learner Modeling (Proceedings from 9th International Conference, AIED 1999)*, pages 51-62.
- [JOVANOVIC2006] Jovanovic, J., Gasevic, D., Brooks, C., Knight, C., Richards, G., and McCalla, G. Ontologies to Support Learning Design Context. From the 1<sup>st</sup> European Conference on Technology Enhanced Learning (EC-TEL 2006), October 1-4, 2006, Crete, Greece.
- [KAY2006] Kay, J. Scrutable Adaptation: Because We Can and Must. In Wade, V. Ashman, H., and Smyth, B. (eds) *Proceedings from the 4<sup>th</sup> International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH2006)*, Dublin, Ireland, June 21-23, 2006, pages 11-19.



[KAY2005] Kay, J., and Lum, A. Exploiting Readily Available Web Data for Scrutable Student Models. In Looi, C., McCalla, G., Bredweg, B. and Breuker, J., editors, *Artificial Intelligence in Education*, Volume 125, pages 338-345.

[KAY2001] Kay, J. Learner Control. In Kobsa, A. (ed) *User Modeling and User-Adapted Interaction: An International Journal*, Issue 11 (2001), pages 111-127.

[KAY1997] Kay, J. Learner Know Thyself: Student Models to Give Learner Control and Responsibility. Invited Keynote address, in Z Halim, T Ottomann, Z Razak (eds), *ICCE'97 International Conference on Computers in Education*, Association for the Advancement of Computing in Education, 1997, pp 17-24.

[KOBASA2002] Kobsa, A. Personalized hypermedia and international privacy. *Communications of the ACM*, Volume 45(5), 2002, pages 64-67.

[KOBASA2001] Kobsa, A. Tailoring Privacy to Users' Needs. In Bauer, M., Gmytrasiwicz, P., and Vassileva, J., editors, *User Modeling 2001 (Proceedings from 8th International Conference, UM 2001)*, pages 303-313.

[KOBASA1990] Kobsa, A. User Modeling in Dialog Systems: Potentials and Hazards. *In AI & Society*, 1990, pages 214-240. Also available online: <http://www1.ics.uci.edu/~kobsa/>.

[KOSBA2005] Kosba, E., Dimitrova, V., Boyle, R. The Evaluation of an Intelligent Teacher Advisor for Web Distance Environments. In Looi, C., McCalla, G., Bredweg, B. and Breuker, J., editors, *Artificial Intelligence in Education*, Volume 125, pages 369-377.

[LINTON2000] Linton, F., and Schaefer, H. Recommender Systems for Learning: Building User and Expert Models through Long-Term Observation of Application Use, *User Modelling and User-Adapted Interaction*, Volume 10, pages 181-207.

[MABBOTT2007] Mabbott, A., and Bull, S. Comparing Student-Constructed Open Learner Model Presentations to the Domain. In Luckin, R., Koedinger, K., and Greer, J., editors, *Artificial Intelligence in Education: Building Technology Rich Learning Contexts that Work*, Volume 158, 2007, pages 281-288.

[MISLEVY1995] Mislevy, R.J., and Gitomer, D.H. The Role of Probability-Based Inference in an Intelligent Tutoring System. In Kobsa, A., editor, *User Modeling and User-Adapted Interaction: An International Journal*, Volume 5, 1995. Kluwer Academic Publishers, Netherlands, pages 253-282.

[MITROVIC2007] Mitrovic, A., and Martin, B. Evaluating the Effect of Open Student Models on Self-Assessment. In Brna, P., editor, *International Journal of Artificial*

*Intelligence in Education Special Issue: Open Learner Models: Research Questions*, Volume 17(2), 2007, pages 121-144.

[MITROVIC2002] Mitrovic, A., and Martin, B. Evaluating the Effects of Open Student Models on Learning. In: P. de Bra, P. Brusilovsky and R. Conejo (eds) *Proceedings of the 2nd Int. Conf on Adaptive Hypermedia and Adaptive Web-based Systems*, AH 2002, Malaga Spain, LCNS 2347, 296-305, 2002.

[MCCALLA2004] McCalla, G. The ecological approach to the design of e-learning environments: purpose-based capture and use of information about learners. In Anderson, T., and Whitelock, D. (eds), *Journal of Interactive Media in Education (JIME)*, Special Issue on The Educational Semantic Web, Volume 1, pages 18-41.

[MCCALLA2000] McCalla, G., Vassileva, J., Greer, J., and Bull, S. Active Learner Modelling. . In Gauthier, G., Frasson, C., and VanLehn, K., eds, *Proceedings from Intelligent Tutoring Systems: 5<sup>th</sup> International Conference*, ITS 2000, Montreal, Canada, June 19-23, 2000, Heidelberg, Germany: Springer-Verlag, pages 53-62.

[MCCALLA1994] McCalla, G., and Greer, J. Granularity-based Reasoning and Belief Revision in Student Models. In Greer, J. E. and McCalla, G. I. (eds), *Student Modelling: The Key to Individualized Knowledge-Based Instruction*, Springer-Verlag, Berlin, 1994, pages 39-62.

[MOODLE2007] Moodle: A Free, Open Source Content Management System for Online Learning. Available online at <http://moodle.org/>. Last accessed August 26, 2007.

[MORALES1999a] Morales, R., Pain, H., Bull, S. and Kay, J. Introduction. In Morales, R., editor, *Workshop on Open, Interactive, and Other Overt Approaches to Learner Modeling (Proceedings from 9th International Conference, AIED 1999)*, pages v-vii.

[MORALES1999b] Morales, R., Pain, H., Conlon, T. From Behaviour to Understandable Presentations of Learner Models: A Case Study. In Morales, R., editor, *Workshop on Open, Interactive, and Other Overt Approaches to Learner Modeling (Proceedings from 9th International Conference, AIED 1999)*, pages 15-24.

[NIU2003] Niu, X., McCalla, G., and Vassileva, J. Purpose-based user modeling in a multi-agent portfolio management system. *Proceedings of UM 2003: International Conference on User Modeling*, Johnstown, Pennsylvania, pages 398-402.

[RUEDA2003] Rueda, U., Larranaga, M. Ferrero, B., Arruarte, A., Elorriaga, J.A. Study of Graphical Issues in a Tool for Dynamically Visualising Student Models. *In the Proceedings of the Workshop on Learner Modelling for Reflection, 11<sup>th</sup> International Conference on Artificial Intelligence in Education, AIED2003*, pages 268-177.

[SELF1994] Self, J. Formal Approaches to Student Modelling. In McCalla, G., and Greer, J. (eds). *Student Modelling: the key to individualized knowledge-based instruction*. Berlin: Springer-Verlag, 1994, pages 295-352.

[SUN2007] Java Reflection. Available online at <http://java.sun.com/javase/6/docs/technotes/guides/reflection/index.html>. Last accessed July 13, 2007.

[TONGCHAI2005] Tongchai, N., and Brna, P. Enhancing Metacognitive Skills through the use of a Group Open Learner Model based on the Zone of Proximal Development. In Kay, J., Lum, A., and Zapata-Rivera, D., editors, *Learner Modelling for Reflection to Support Learner Control, Metacognition, and Improved Communications*, AIED Workshop 11, Amsterdam, July 18-22, 2005, pages 91-99..

[VASSILEVA2003] Vassileva, J., McCalla, G., and Greer, J. Multi-agent Multi-user Modelling, *User Modelling and User Adapted Interaction*, Issue 13, 2003, pages 1-31.

[VASSILEVA1999] Vassileva, J., Greer, J., McCalla, G. Openness and Disclosure in Multi-Agent Learner Models. In Morales, R., editor, *Workshop on Open, Interactive, and Other Overt Approaches to Learner Modeling (Proceedings from 9th International Conference, AIED 1999)*, pages 43-49.

[WAERN2004] Waern, A. User involvement in automatic filtering – an experimental study. *User Modelling and User Adapted Interaction*, Volume 14, pages 2001-237.

[WEBCT2007] WebCT. Available online at: <http://www.webct.com/>. Last Accessed August 26, 2007.

[WEBER2001] Weber, G., and Brusilovsky, P. ELM-ART: An Adaptive Versatile System for Web-Based Instruction, *International Journal of Artificial Intelligence in Education*, 12 (4), pages 351-384.

[ZAPATA2005] Zapata-Rivera, D., Underwood, J., Bauer, M. Advanced Reporting Systems in Assessment Environments. In Kay, J., Lum, A., and Zapata-Rivera, D., editors, *Learner Modelling for Reflection to Support Learner Control, Metacognition, and Improved Communications*, AIED Workshop 11, Amsterdam, July 18-22, 2005, pages 23-31.

[ZAPATA2004] Zapata-Rivera, J., and Greer, J.E. Interacting with Inspectable Bayesian Student Models, *International Journal of Artificial Intelligence in Education*, 14 (2), pages 127-163.

[ZAPATA2003] Zapata-Rivera, J., and Greer, J. Analyzing Student Reflection in The Learning Game. In Aleven, V. et al, eds, *Proceedings from AIED2003*, University of Sidney, pages 288-298.

[ZAPATA2000] Zapata-Rivera, J., and Greer, J.E. Inspecting and Visualizing Distributed Bayesian Student Models. . In Gauthier, G., Frasson, C., and VanLehn, K., eds, *Proceedings from Intelligent Tutoring Systems: 5<sup>th</sup> International Conference*, ITS 2000, Montreal, Canada, June 19-23, 2000, Heidelberg, Germany: Springer-Verlag, pages 544-553.

## APPENDIX A: CHARACTERISTICS AVAILABLE IN QUERY TOOL

This appendix contains a list of characteristics available in the Query Tool/Compare Progress feature of iHelp Courses. The characteristics are listed as methods of the Data class (see Appendix C for the system design of the Query Tool). The list contains the return type of the method, the name of the method, and the method parameters.

Note that each method has the parameters “Learner” and “Viewer”. The “Learner” is the subject of the open model (whose model is being viewed). The “Viewer” is the user currently viewing the learner model. The methods calculate different results based on who the learner is and who the viewer is, or completely block access to the method if the viewer does not have the appropriate permission to view the information.

### **Table accessTimesInLastWeek(Learner learner, Viewer viewer)**

Returns a table of the times the learner has accessed the course in the last week. The columns in the table are divided into columns representing three hour divisions of the week. If the learner accessed the course during the three hour division, the cell’s background color is yellow, otherwise the background color is the default for the table.

### **Table accessTimesSince(Learner learner, Viewer viewer, Date sinceDate)**

Returns a table of the times the learner has accessed the course since the specified date. The columns in the table are divided into columns representing three hour divisions of the entire time. If the learner accessed the course during the three hour division, the cell’s background color is yellow, otherwise the background color is the default for the table.

### **ArrayListParameter<Date> allContentAccessedDatesInCourse(Learner learner, Viewer viewer, Course course)**

Returns a list of dates/times when the learner accessed content in the specified course.

**ArrayListParameter<Date> allContentAccessedDatesInCourseSince(Learner learner, Viewer viewer, Course course, Date date)**

Returns a list of dates/times when the learner accessed content in the specified course where the time is after the specified date.

**ArrayListParameter<String> allContentTitlesViewedInCourse(Learner learner, Viewer viewer, Course course)**

Returns a list of all the content titles the learner has viewed in the course.

**ArrayListParameter<Date> allLogins(Learner learner, Viewer viewer)**

Returns a list of all the dates/times the learner has logged into the iHelp Courses system.

**ArrayListParameter<Date> allLoginsSince(Learner learner, Viewer viewer, Date date)**

Returns a list of all the dates/times the learner has logged into the iHelp Courses system after the specified date.

**ArrayListParameter<Table> allQuizQuestionsAndAnswers(Learner learner, Viewer viewer, Quiz quiz)**

Returns a list of tables with the questions, scores, the learner's answers, and the feedback given to the learner based on their responses for the specified quiz.

**TimeSpent averagePostingReadLagTimeInCategory(Learner learner, Viewer viewer, Category category)**

Returns the average time a learner takes to read a new message posted in the specified category.

**TimeSpent averagePostingReadLagTimeInCourse(Learner learner, Viewer viewer, Course course)**

Returns the average time a learner takes to read a new message posted in any category assigned to the course.

**TimeSpent averagePostingReplyLagTimeInCategory(Learner learner, Viewer viewer, Category category)**

Returns the average time a learner takes to respond to a message posted in the specified category.

**TimeSpent averagePostingReplyLagTimeInCourse(Learner learner, Viewer viewer, Course course)**

Returns the average time a learner takes to respond to a message posted in any category assigned to the course.

**float averageScoreOnQuiz(Learner learner, Viewer viewer, Quiz quiz)**

Returns the average score a learner has achieved on a quiz compiled from all the learner's attempts on the quiz.

**String email(Learner learner, Viewer viewer)**

Returns the learner's email address.

**Date firstContentAccessedEnterTime(Learner learner, Viewer viewer, Content content)**

Returns the date/time of the first time the learner accessed the specified content.

**Date firstContentAccessedExitTime(Learner learner, Viewer viewer, Content content)**

Returns the date/time when the learner exited the page of the first time the learner accessed the specified content.

**Date firstLoginDate(Learner learner, Viewer viewer)**

Returns the date/time when the learner first logged on to the iHelp Course system.

**String firstName(Learner learner, Viewer viewer)**

Returns the learner's first name.

**Date firstTimeUserTookQuiz(Learner learner, Viewer viewer, Quiz quiz)**

Returns the date/time the learner first took the specified quiz.

**boolean hasRole(Learner learner, Viewer viewer, Role role)**

Returns true if the learner has the specified role, and false otherwise.

**float highestScoreOnQuiz(Learner learner, Viewer viewer, Quiz quiz)**

Returns the learner's highest score on the quiz based on all the learner's attempts for this quiz.

**Date lastCategoryReadDateInCourse(Learner learner, Viewer viewer, Course course)**

Returns the date/time the learner last read a message in one of the categories associated with the course.

**Date lastContentAccessedDate(Learner learner, Viewer viewer)**

Returns the date/time the learner last accessed material in the iHelp Courses system.

**Date lastContentAccessedDateInCourse(Learner learner, Viewer viewer, Course course)**

Returns the date/time the learner last accessed material in the specified course.

**Date lastContentAccessedEnterTime(Learner learner, Viewer viewer, Content content)**

Returns the date/time the learner last started viewing the specified content.

**Date lastContentAccessedExitTime(Learner learner, Viewer viewer, Content content)**

Returns the date/time the learner left the page of the specified content.

**Date lastLoginDate(Learner learner, Viewer viewer)**

Returns the date/time the learner last logged on to the iHelp Courses system.

**Date lastMessageDateInChannel(Learner learner, Viewer viewer, Channel channel)**

Returns the date/time the learner last sent a message in the synchronous chat environment (iHelp Chat) in the specified channel.

**Date lastMessageDateInCourse(Learner learner, Viewer viewer, Course course)**

Returns the date/time the learner last sent a message in iHelp Chat for any channel associated with the specified course.

**String lastName(Learner learner, Viewer viewer)**

Returns the last name of the learner.

**Date lastPostDateInCategory(Learner learner, Viewer viewer, Category category)**

Returns the last date/time the learner posted a message in the specified category.

**Table lastQuizQuestionsAndAnswers(Learner learner, Viewer viewer, Quiz quiz)**

Returns a table of the questions, scores, the learner's answers, and the feedback given to the learner based on their responses for the last time the learner took the specified quiz.

**Date lastReadDateInCategory(Learner learner, Viewer viewer, Category category)**

Returns the last date/time the learner read a message in the specified category.

**Date lastTimeUserTookQuiz(Learner learner, Viewer viewer, Quiz quiz)**

Returns the last date/time the learner took the specified quiz.

**float lowestScoreOnQuiz(Learner learner, Viewer viewer, Quiz quiz)**

Returns the lowest score the learner achieved on the specified quiz.

**String name(Learner learner, Viewer viewer)**

Returns the full name (first name, last name) of the learner.

**int numberLogins(Learner learner, Viewer viewer)**

Returns the number of logins the learner has had in the iHelp Courses system.



**int numberLoginsInPastWeek(Learner learner, Viewer viewer)**

Returns the number of logins the learner has had in the iHelp Courses system in the last week.

**int numberLoginsSince(Learner learner, Viewer viewer, Date date)**

Returns the number of logins the learner has had in the iHelp Courses system since the specified date.

**int numberMessagesInChannel(Learner learner, Viewer viewer, Channel channel)**

Returns the number of messages total the learner has sent to the synchronous chat (iHelp Chat) in the specified channel.

**String numberMessagesInChannelComparedToTotal(Learner learner, Viewer viewer, Channel channel)**

Returns a percentage of the messages the learner has sent in the specified channel as compared to the total number of messages in that channel.

**int numberMessagesInCourse(Learner learner, Viewer viewer, Course course)**

Returns the number of messages the learner has sent in any channel associated with the specified course.

**int numberPostsMadeInCategory(Learner learner, Viewer viewer, Category category)**

Returns the number of posts the learner has made in the asynchronous discussion forum (iHelp Discussions) in the specified category.

**int numberPostsMadeInCategoryBefore(Learner learner, Viewer viewer, Category category, Date date)**

Returns the number of posts the learner has made in the specified category before the specified date.

**int numberPostsMadeInCourse(Learner learner, Viewer viewer, Course course)**

Returns the number of posts the learner has made in any category associated with the specified course.

**int numberPostsMadeInCourseBefore(Learner learner, Viewer viewer, Course course, Date date)**

Returns the number of posts the learner has made in any category associated with the specified course before the specified date.

**int numberPostsReadInCategory(Learner learner, Viewer viewer, Category category)**

Returns the number of posts the learner has read in the specified category.

**int numberPostsReadInCategoryBefore(Learner learner, Viewer viewer, Category category, Date date)**

Returns the number of posts the learner has read in the specified category before the specified date.

**int numberPostsReadInCourse(Learner learner, Viewer viewer, Course course)**

Returns the number of posts the learner has read in any category associated with the course.

**int numberPostsReadInCourseBefore(Learner learner, Viewer viewer, Course course, Date date)**

Returns the number of posts the learner has read in any category associated with the course before the specified date.

**int numberQuizzesCompletedInCourse(Learner learner, Viewer viewer, Course course)**

Returns the number of quizzes the learner has completed for the course.

**int numberThreadsStartedInCategory(Learner learner, Viewer viewer, Category category)**

Returns the number of “top level” threads the learner has started in the specified category.

**int numberThreadsStartedInCategoryBefore(Learner learner, Viewer viewer, Category category, Date date)**

Returns the number of “top level” threads the learner has started in the specified category before the specified date.

**int numberThreadsStartedInCourse(Learner learner, Viewer viewer, Course course)**

Returns the number of “top level” threads the learner has started in any category associated with the specified course.

**int numberThreadsStartedInCourseBefore(Learner learner, Viewer viewer, Course course, Date date)**

Returns the number of “top level” threads the learner has started in any category associated with the specified course before the specified date.

**int numberTimesAccessedContent(Learner learner, Viewer viewer, Content content)**

Returns the number of time the learner has accessed the specified content.

**int numberTimesTookQuiz(Learner learner, Viewer viewer, Quiz quiz)**

Returns the number of times the learner has taken the specified quiz.

**Number participationCmpt100Online(Learner learner, Viewer viewer, Course course)**

Returns the participation (as calculated for CMPT 100 Online) for the specified course. The participation is calculated using the following equations:

$$\text{Participation} = 0.7 \times P_m + 0.3 P_r \quad (\text{A.1})$$

where  $P_m$  = the percentage of posts the learner made (Equation A.2) to a maximum of 100%

and  $P_r$  = the percentage of posts the learner read (Equation A.3) to a maximum of 100%

$$P_m = N_m / A_m \times 100\% \quad (\text{A.2})$$

where  $P_m$  = the percentage of posts the learner made

and  $N_m$  = the number of posts the learner made

and  $A_m$  = the average number of posts made by peers

$$P_r = N_r / N_t \times 100\% \quad (\text{A.3})$$

where  $P_r$  = the percentage of posts the learner read

and  $N_r$  = the number of posts the learner read

and  $N_t$  = the total number of posts

**Number participationCmpt100OnlineOutOf10(Learner learner, Viewer viewer, Course course)**

Returns the participation (as calculated for CMPT 100 Online) for the specified course as a mark out of 10.

**float percentagePostsReadInCategory(Learner learner, Viewer viewer, Category category)**

Returns the percentage of posts the learner has read in the specified category based on the total number of posts in the category.

**float percentagePostsReadInCourse(Learner learner, Viewer viewer, Course course)**

Returns the percentage of posts the learner has read in any category associated with the specified course based on the total number of posts for the course.

**String preferenceToRunFromCd(Learner learner, Viewer viewer)**

Returns the learner's preference to run the course material from the course cd.

**ArrayListParameter<Quiz> quizMarksInCourse(Learner learner, Viewer viewer, Course course)**

Returns a list of quizzes (along with the learner's score on the quiz) for all quiz attempts in the specified course.

**ArrayListParameter<Quiz> quizzesCompletedInCourse(Learner learner, Viewer viewer, Course course)**

Returns a list of quizzes that the learner has attempted in the specified course.

**ArrayListParameter<Quiz> quizzesNotCompletedInCourse(Learner learner, Viewer viewer, Course course)**

Returns a list of quizzes that the learner has not attempted in the specified course.

**boolean recentlyLoggedIn(Learner learner, Viewer viewer)**

Returns true if the learner has logged on to the iHelp Courses System in the last ten minutes, and false otherwise.

**ArrayListParameter<Float> scoresOnQuiz(Learner learner, Viewer viewer, Quiz quiz)**

Returns a list of the learner's scores for the specific quiz.

**TimeSpent timeSpentOnContent(Learner learner, Viewer viewer, Content content)**

Returns the amount of time the learner has spent on the specified content. The time does not include the time spent on any sub content.

**TimeSpent timeSpentOnContentAndChildren(Learner learner, Viewer viewer, Content content)**

Returns the amount of time the learner has spent on the specified content. The time includes the time spent on any sub content.

**ArrayListParameter<Date> timesUserTookQuiz(Learner learner, Viewer viewer, Quiz quiz)**

Returns a list of dates/times the learner attempted the specified quiz.

**String userid(Learner learner, Viewer viewer)**

Returns the learner's user id.

**ArrayListParameter<String> usualLoginDaysOfTheWeek(Learner learner, Viewer viewer)**

Returns a list of the learner's typical day(s) of the week they log into the iHelp Courses system.

**ArrayListParameter<String> usualLoginTimesOfTheDay(Learner learner, Viewer viewer)**

Returns a list of the learner's typical time(s) of the day they log into the iHelp Courses system.

**boolean viewed(Learner learner, Viewer viewer, Content content)**

Returns true if the learner has viewed the specified content, and false otherwise.

## APPENDIX B: FILTERS AVAILABLE IN QUERY TOOL

This appendix contains a list of filters available in the Query Tool/Compare Progress feature of iHelp Courses. The filters are listed as methods of the Filter class (see Appendix C for the system design of the Query Tool). The list contains the return type of the method (a list of learners), the name of the method, and the required parameters.

Each method allows the instructor to filter learners IN (meaning that learners matching the filter remain in the output learner list) or OUT (meaning that learners matching the filter are removed from the output learner list). This is specified by the `FilterType` parameter in each of the methods.

There are two different types of comparisons used by the Filter methods below: `StringComparison` and `NumberComparison`. A `StringComparison` allows the instructor to specify additional details on how to compare the strings by providing additional parameters. For example, the instructor may choose to compare the strings to see if the string contains the search string, starts with the search string, or equals the search string. A `NumberComparison` allows the instructor to specify additional details on how to compare two numbers. For example, the instructor may choose to compare the numbers to see if the number is equal to, greater than, or less than the other number.

Each of the methods below has a parameter “learners” (with the exception of the `defaultLearners` method). The filter will start with the learners sent into the method by this parameter and filter learners IN or OUT based on the `FilterType`. This allows the output of one filter to be the input of another filter. The `defaultLearners` method

returns a list of learners in the specified roles, and is the starting point for the filters methods, as it returns the largest set of learners before the filters will be applied.

**ArrayListParameter<Learner> accessedCourseRecently(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners, Course course)**

If the FilterType is IN, returns a list of learners who have accessed the specified course in the last ten minutes. If the FilterType is OUT, returns a list of learners who have not accessed the course in the last ten minutes. Accessing the course means either viewing a content page in the course or logging on to the iHelp Courses system.

**ArrayListParameter<Learner> accessedCourseSince(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners, Course course, Date date)**

If the FilterType is IN, returns a list of learners who have accessed the specified course since the specified date. If the FilterType is OUT, returns a list of learners who have not accessed the course since the specified date.

**ArrayListParameter<Learner> defaultLearners(ArrayList<Role> roles)**

Returns a list of learners who have one or more of the specified roles.

**ArrayListParameter<Learner> firstName(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners, StringComparison comparison, String firstname)**

If the FilterType is IN, returns a list of learners who match the comparison of their first names. If the FilterType is OUT, returns a list of learners who do not match the comparison of their first names.

**ArrayListParameter<Learner> hasRole(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners, Role role)**

If the FilterType is IN, returns a list of learner who have the specified role. If the FilterType is OUT, returns a list of learners who do not have the specified role.

**ArrayListParameter<Learner> lastName(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners, StringComparison comparison, String lastname)**

If the FilterType is IN, returns a list of learner who match the comparison of their last names. If the FilterType is OUT, returns a list of learners who do not match the comparison of their last names.

**ArrayListParameter<Learner> loggedInSince(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners, Date date)**

If the FilterType is IN, returns a list of learners who have logged on to the iHelp Courses system since the specified date. If the FilterType is OUT, returns a list of

learners who have not logged on to the iHelp Courses system since the specified date.

**ArrayListParameter<Learner> numberTimesTookQuiz(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners, Quiz quiz, NumberComparison comparison, Number number)**

If the FilterType is IN, returns a list of learners who have completed the specified quiz the specified number of times. If the FilterType is OUT, returns a list of learners who have not completed the specified quiz the number of specified times.

**ArrayListParameter<Learner> recentlyLoggedIn(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners)**

If the FilterType is IN, returns a list of learners who have logged on to the iHelp Courses system in the last ten minutes. If the FilterType is OUT, returns a list of learners who have not logged on to the iHelp Courses system in the last ten minutes.

**ArrayListParameter<Learner> userid(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners, StringComparison comparison, String userid)**

If the FilterType is IN, returns a list of learners who match the comparison of their userid's. If the FilterType is OUT, returns a list of learners who do not match the comparison of their userid's.

**ArrayListParameter<Learner> viewed(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners, Content content)**

If the FilterType is IN, returns a list of learners who have viewed the specified content. If the FilterType is OUT, returns a list of learners who have not viewed the specified content.

**ArrayListParameter<Learner> viewerOfPage(Viewer viewer, FilterType filterType, ArrayListParameter<Learner> learners)**

If the FilterType is IN, returns the learner who is currently viewing the learner model. If the FilterType is OUT, returns a list of learners who are not currently viewing the learner model.

## APPENDIX C: QUERY TOOL SYSTEM DESIGN

The Query Tool uses Java Reflection [SUN2007] to create the appropriate learner models when the view is generated. Reflection is used throughout the entire process as explained in detail here.

When the instructor requests the View Creation Interface page, the Data and Filter class definitions are scanned to find the methods available to the Query Tool and the required parameters of the methods. The interface is then built based on the appropriate methods. Therefore, when new methods are added to (or removed from) the Data and Filter classes, the interface is automatically updated to reflect the changes.

The instructor uses a DHTML web interface to create the view to include the desired characteristics and appropriate filters. If one of the characteristics requires additional parameters, parameter boxes appear on the screen asking for further details. When the instructor has specified all the required information (including a descriptive purpose for the learner model and a list of roles who can also generate the view) and submits the interface, an XML message is sent via AJAX [GARRET2005] to the server with the view template.

The server parses the XML message, finds the appropriate characteristics, and creates two Java classes based on the parameters specified by the instructor – a “View”, and a “Model” which will be used when the view is generated/requested by the user. The “View” class will eventually generate the entire view by finding the list of learners the instructor has requested, and then generating each of the learners’ models (using the Model class). The “Model” represents the “rows” of the view –



essentially a model is created for each learner in the view. The “Model” class contains methods to find each of the characteristics based on the view template created by the instructor. These two classes are compiled and saved as bytecode to the database. The instructor’s purpose and the appropriate access to the view (i.e. who can generate it) are also saved into the database. Another AJAX message is sent back to the instructor indicating the success or failure of this process.

When the user first clicks on the List of Views, the database is queried to retrieve a list of appropriate views for the user for that course, along with an indication of who created the view and an expected time for the view to generate. When the user selects a view to generate, the View class is loaded from the bytecode retrieved from the database and invoked using Java Reflection, passing in the appropriate run time parameters such as the Viewer, course, etc. The View first finds the default set of learners (for example, the learners in the course, or the learners in the course and the teaching assistants, etc), then uses filters from the Filter class (see Appendix B) to find a refined set of learners. Once the set of learners has been found, the View then generates the models for each of the learners using the appropriate Model. The Model uses data methods from the Data class (see Appendix A) to calculate each characteristic, and all characteristics are generated at run time – no offline calculations are completed. Once all of the models have been generated, the View requests each model to return a display of the model, which for the default table view is an HTML table row with the learner’s characteristics in table cells. The View gathers each of these rows, and returns an HTML table to finally be displayed. This process allows other types of views to be quickly implemented, as methods just have to be created to generate the appropriate View and Model classes, since the interface to each (generate and display methods) are common to each of the views, and HTML is the output of the display method.

Every time a user generates a learner model, the time the model took to generate is saved to the database. The generation time is used to calculate the expected time to

generate the model, and is displayed on the list of views page. Usage data is also stored to identify who generated the learner model.

The Query Tool (Compare Progress feature) is dependent on the iHelp Courses system for certain information. Only instructors and administrators specified by the system's edit course page can create views for the course. Only those roles associated with each course on the edit course page will show up in the default learner role selection list (to keep unauthorized users from viewing information in other courses). The Query Tool also requires a course to be specified in order for views to be created (in other words, the Query Tool cannot be used at a system level to generate views across courses). Links to the iHelp Discussions and iHelp Chat systems can be made through the iHelp Courses system if links to categories and channels are made in the course. However, the Query Tool cannot be used to generate statistics for these courses without a course set up in iHelp Courses.

Figure C-1 shows a test model created using the Query Tool to highlight this process. Figure C-2 is a code listing of the View class created for the model. Figure C-3 is a code listing of the Model class created for the view. These classes are produced by the system when the instructor creates a new view.

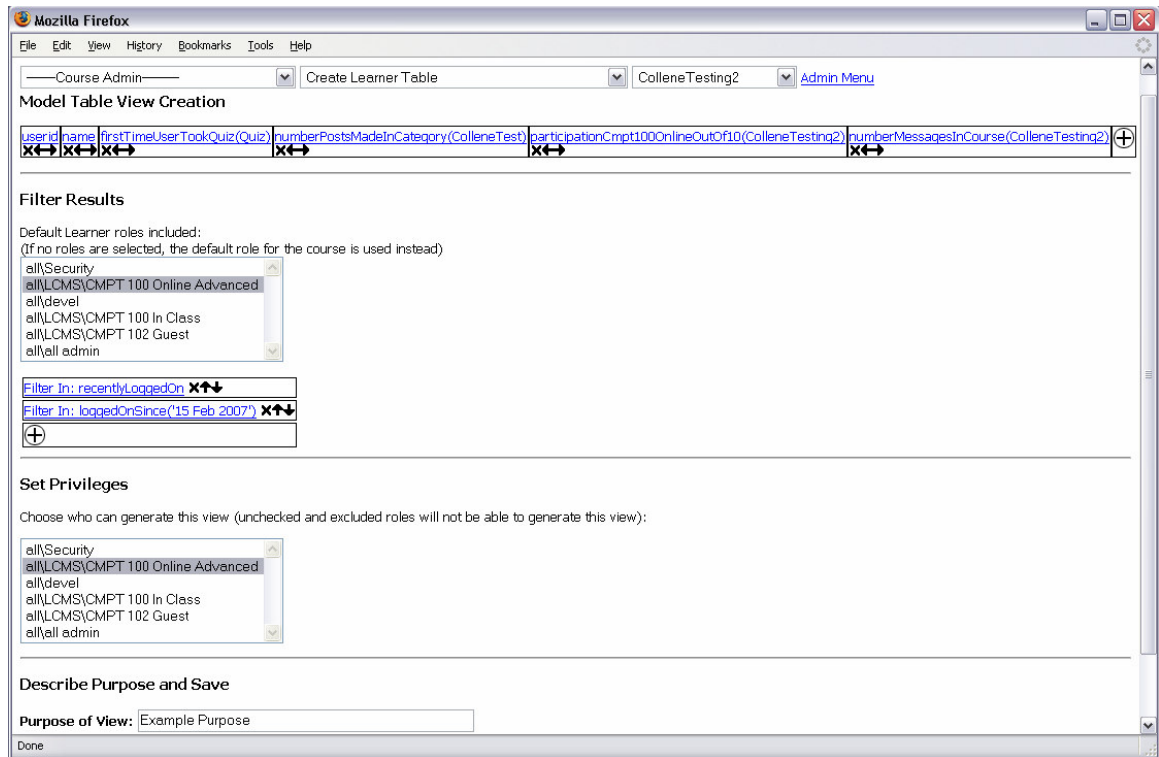


Figure C-1: Example purpose created using Query Tool interface

```

package ca.usask.cs.lcas.stats.runtime;
import ca.usask.cs.lcas.stats3.*;
import ca.usask.cs.lcas.stats3.parameters.*;
import java.util.ArrayList;
import java.util.Iterator;
import ca.usask.cs.lcas.OperationFailedException;
import ca.usask.cs.lcas.stats3.BlockedPermissionException;
public class View17680 implements IView{
    private int _modelid=100;
    private int _courseid=2;

    public String display(Viewer viewer) throws OperationFailedException{
        String str="<table id='tblSort'>";
        str+ "<thead>";

        str+ "<tr><th><a href='\"javascript:sortTable(\"tblSort\",0)\">\">UserId()</a></th>"
            + "<th><a href='\"javascript:sortTable(\"tblSort\",1)\">\">Name()</a></th>"
            + "<th><a href='\"javascript:sortTable(\"tblSort\",2,\"date\")\">\">First Time User Took Quiz(Quiz)</a></th>"
            + "<th><a href='\"javascript:sortTable(\"tblSort\",3,\"int\")\">\">Number Posts Made In Category(ColleneTest)</a></th>"
            + "<th><a href='\"javascript:sortTable(\"tblSort\",4,\"float\")\">\">Participation Cpt100 Online Out Of10(ColleneTesting2)</a></th>"
            + "<th><a href='\"javascript:sortTable(\"tblSort\",5,\"int\")\">\">Number Messages In Course(ColleneTesting2)</a></th></tr>";
        str+ "</thead>";

        String filterResultsString="";
        ArrayList<String> filterResults=new ArrayList<String>();
        ArrayList<Parameter<Learner>> learners=filter(viewer,filterResults);
        ArrayList<Model> models=generate(learners,viewer);
        if(models.size()==0){
            str+ "<tr><td colspan='7'>No learners match your criteria</td></tr>";
        }
        for(Iterator<Model> i=models.iterator();i.hasNext();){
            str+ i.next().display();
        }
        for(Iterator<String> it=filterResults.iterator();it.hasNext();){
            filterResultsString+ it.next()+ "<br />";
        }

        return str+ "</table><p class='info'>"+ learners.size()+ " learners in table</p><p class='warning'>"+ filterResultsString+ "</p>";
    }

    private ArrayList<Parameter<Learner>> filter(Viewer viewer, ArrayList<String> filterResults) throws OperationFailedException{
        ArrayList<Role> roles=new ArrayList<Role>();
        roles.add(new Role(296));
        ArrayList<Parameter<Learner>> learners=ca.usask.cs.lcas.stats3.Filter.defaultLearners(roles);
        try{
            learners=ca.usask.cs.lcas.stats3.Filter.recentlyLoggedInOn(viewer,ca.usask.cs.lcas.stats3.Filter.FilterType.IN,learners);
        }catch(BlockedPermissionException bpe){
            filterResults.add("Filter <b><i>recentlyLoggedInOn</i></b> was not applied because you do not have permissions to run the filter.");
        }catch(java.lang.NoSuchMethodError nme){
            filterResults.add("Filter <b><i>recentlyLoggedInOn</i></b> was not applied because the filter no longer exists.");
        }catch(Exception e){
            filterResults.add("Filter <b><i>recentlyLoggedInOn</i></b> was not applied because of an internal server error.");
        }

        try{
            learners=ca.usask.cs.lcas.stats3.Filter.loggedOnSince(viewer,ca.usask.cs.lcas.stats3.Filter.FilterType.IN,learners,new Date("Feb15|2007"));
        }catch(BlockedPermissionException bpe){
            filterResults.add("Filter <b><i>loggedOnSince</i></b> was not applied because you do not have permissions to run the filter.");
        }catch(java.lang.NoSuchMethodError nme){
            filterResults.add("Filter <b><i>loggedOnSince</i></b> was not applied because the filter no longer exists.");
        }catch(Exception e){
            filterResults.add("Filter <b><i>loggedOnSince</i></b> was not applied because of an internal server error.");
        }

        return learners;
    }

    private ArrayList<Model> generate(ArrayList<Parameter<Learner>> learners, Viewer viewer) throws OperationFailedException
    {
        ArrayList<Model> models=new ArrayList<Model>();
        for(Iterator<Learner> i=learners.iterator();i.hasNext();){
            models.add(ModelManager.getInstance().generateModel(_modelid,i.next().getUserId(),viewer._courseid));
        }
        return models;
    }
}

```

Figure C-2: Code listing of the view for the example purpose

```

package ca.usask.cs.lcas.stats.runtime;
import ca.usask.cs.lcas.stats3.*;
import ca.usask.cs.lcas.stats3.parameters.*;
import java.util.ArrayList;
import java.util.Iterator;
import ca.usask.cs.lcas.OperationFailedException;
import ca.usask.cs.lcas.stats3.BlockedPermissionException;
public class View17680 implements IView{
    private int _modelid=100;
    private int _courseid=2;

    public String display(Viewer viewer) throws OperationFailedException{
        String str="<table id='tblSort'>";
        str+ "<thead>";

        str+ "<tr><th><a href='\"javascript:sortTable('tblSort',0)\">Userid()</a></th>"
            + "<th><a href='\"javascript:sortTable('tblSort',1)\">Name()</a></th>"
            + "<th><a href='\"javascript:sortTable('tblSort',2,'date')\">First Time User Took Quiz(Quiz)</a></th>"
            + "<th><a href='\"javascript:sortTable('tblSort',3,'int')\">Number Posts Made In Category(ColleneTest)</a></th>"
            + "<th><a href='\"javascript:sortTable('tblSort',4,'float')\">Participation Cpt100 Online Out Of10(ColleneTesting2)</a></th>"
            + "<th><a href='\"javascript:sortTable('tblSort',5,'int')\">Number Messages In Course(ColleneTesting2)</a></th></tr>";
        str+ "</thead>";

        String filterResultsString="";
        ArrayList<String> filterResults=new ArrayList<String>();
        ArrayList<Parameter<Learner>> learners=filter(viewer,filterResults);
        ArrayList<Model> models=generate(learners,viewer);
        if(models.size()==0){
            str+ "<tr><td colspan='7'>No learners match your criteria</td></tr>";
        }
        for(Iterator<Model> i=models.iterator();i.hasNext();){
            str+ i.next().display();
        }
        for(Iterator<String> it=filterResults.iterator();it.hasNext();){
            filterResultsString+ it.next()+ "<br />";
        }

        return str+ "</table><p class='info'>" + learners.size()+ " learners in table</p><p class='warning'>" + filterResultsString+ "</p>";
    }

    private ArrayList<Parameter<Learner>> filter(Viewer viewer, ArrayList<String> filterResults) throws OperationFailedException{
        ArrayList<Role> roles=new ArrayList<Role>();
        roles.add(new Role(296));
        ArrayList<Parameter<Learner>> learners=ca.usask.cs.lcas.stats3.Filter.defaultLearners(roles);
        try{
            learners=ca.usask.cs.lcas.stats3.Filter.recentlyLoggedInOn(viewer,ca.usask.cs.lcas.stats3.Filter.FilterType.IN,learners);
        }catch(BlockedPermissionException bpe){
            filterResults.add("Filter <b><i>recentlyLoggedInOn</i></b> was not applied because you do not have permissions to run the filter.");
        }catch(java.lang.NoSuchMethodError nme){
            filterResults.add("Filter <b><i>recentlyLoggedInOn</i></b> was not applied because the filter no longer exists.");
        }catch(Exception e){
            filterResults.add("Filter <b><i>recentlyLoggedInOn</i></b> was not applied because of an internal server error.");
        }

        try{
            learners=ca.usask.cs.lcas.stats3.Filter.loggedOnSince(viewer,ca.usask.cs.lcas.stats3.Filter.FilterType.IN,learners,new Date("Feb15|2007"));
        }catch(BlockedPermissionException bpe){
            filterResults.add("Filter <b><i>loggedOnSince</i></b> was not applied because you do not have permissions to run the filter.");
        }catch(java.lang.NoSuchMethodError nme){
            filterResults.add("Filter <b><i>loggedOnSince</i></b> was not applied because the filter no longer exists.");
        }catch(Exception e){
            filterResults.add("Filter <b><i>loggedOnSince</i></b> was not applied because of an internal server error.");
        }

        return learners;
    }

    private ArrayList<Model> generate(ArrayList<Parameter<Learner>> learners, Viewer viewer) throws OperationFailedException
    {
        ArrayList<Model> models=new ArrayList<Model>();
        for(Iterator<Learner> i=learners.iterator();i.hasNext();){
            models.add(ModelManager.getInstance().generateModel(_modelid,i.next().getUserid(),viewer._courseid));
        }
        return models;
    }
}

```

Figure C-3: Code listing of the model for the example purpose

## APPENDIX D: INSTRUCTOR CONSENT FORM AND SURVEY

Department of Computer Science

University of Saskatchewan  
Informed Consent Form

Research Project:	Active Open Learner Modelling for Reflection (AOLMR)
-------------------	--

Investigators:	Dr. Gordon McCalla, Professor, Department of Computer Science (966-4902), mccalla@cs.usask.ca
----------------	--

	Collene Hansen, Instructor (CMPT 100 Online), Department of Computer Science (966-8647), collene@cs.usask.ca
--	--

We are investigating the usability and effect of the Compare Progress feature of the iHelp Courses system (as part of the Active Open Learner Modelling for Reflection project, AOLMR). We would like your consent to participate in this study. This consent form should give you the basic idea of what the research is about and what your participation will involve. If you would like more detail about something mentioned here, or information is not included here, please contact one of the study investigators listed above. Please take the time to read this form carefully and to understand any accompanying information.

This study will observe the class interactions and system usage of the Compare Progress feature of the iHelp Courses system (as part of the Active Open Learner Modelling for Reflection project, AOLMR). The Compare Progress feature has been designed as a study tool to help online students reflect on their progress in the course

by comparing their progress and activities in the course compared to their peers. The Compare Progress feature can also be used by instructors in a course to identify how well their students are progressing in the online course. The main goal of the study will be to gain insight into the usability of the Compare Progress feature and its impact on reflection and teaching in an actual class. CMPT 100 (on campus and online sections) and Transforming Teaching were targeted as candidates for this tool because of the class discussion, and online delivery aspects of the class. The Investigators are confident that the class will also benefit from the tool.

The data collected from this study will be used in articles for publication in journals and conference proceedings. As one way of thanking you for your time, we will be pleased to make available to you a summary of the results of this study once they have been compiled. This summary will outline the research and discuss our findings and recommendations.

All of the information we collect from you (data logged by the computer, observations made by the experimenters, and your questionnaire responses) will be stored so that your name, nsid, or email address is not associated with it. Any write-ups of the data will not include any information that can be linked directly to you. The research materials will be stored with complete security throughout the entire investigation.

**By signing this form, you give permission to the Investigators to use examples taken from your interaction with the Compare Progress feature of iHelp Courses. None of your identifying information will be shared with anyone outside of the Investigators. The data in fact will also appear anonymously to the Investigators -- it will not contain information that would be able to link it to you. If you do not sign, we will only be able to use statistical measures of system usage.**

Your continued participation should be as informed as your initial consent, so you should feel free to ask for clarification or new information throughout your participation. If you have further questions concerning matters related to this research, please contact one of the investigators listed above.

By choosing the "Yes, I would like to participate" option below, you indicate that you have understood to your satisfaction the information regarding participation in the research project and agree to participate. In no way does this waive your legal rights nor release the investigators, sponsors, or involved institutions from their legal and professional responsibilities. If you have further questions about this study or your rights as a participant, please contact:

- Dr. Gordon McCalla, Professor Dept. Computer Science (306) 966-4902  
mccalla@cs.usask.ca
- Office of Research Ethics University of Saskatchewan (306) 966-2084

**You are free to opt out at any time and need only inform an Investigator.**

If you would like a copy of this consent form please print it through your browser, or contact one of the investigators listed above. This research has the ethical approval of the Office of Research Services at the University of Saskatchewan.

Date: April 27th, 2007

**Signed:**  
(please enter your full name)

**Email:**  
(for possible contact)

- ☒ Yes, I would like to participate
- ☐ No, I don't want to take part

### iHelp Courses Compare Progress - Instructor Survey

Thank you for your participation in this questionnaire. The responses you provide will be used to evaluate the effectiveness of the Compare Progress feature and modify the tools for future course offerings. Please limit your responses to comments about the Compare Progress feature (to be described in the questionnaire), and not about the iHelp applications in general.

There are several parts to this questionnaire:

- The first part of the questionnaire asks about the two main interfaces of the Compare Progress feature - the View Creation Interface and the List of Views Interface. These questions will ask your opinion on the ease of use of the tools, how it helped your workflow, etc.
- The second part of the questionnaire will ask you about how the Compare Progress feature affected your course (if you changed your teaching strategies based on information you found from the views, and whether it changed your understanding of the learners).
- The third part of the questionnaire asks about privacy in the Compare Progress feature (whether the privacy protection provided is enough, and what levels you feel are appropriate).
- The last part of the questionnaire asks your opinion about additional features for the Compare Progress feature, and leaves space for your comments.

#### **View Creation Interface**

The “View Creation” interface is accessed in the Edit Course page of iHelp Courses (using the “Create Stats Table View for Learners” link). A sample of the screen is shown below:



**Model Table View Creation**

**userid** **Table Headers (characteristics to include in this view)**

<input type="radio"/> Name	<input type="radio"/> Has Role
<input checked="" type="radio"/> Userid	<input type="radio"/> Viewed
<input type="radio"/> Email	<input type="radio"/> First Name
<input type="radio"/> Last Name	<input type="radio"/> Recently Logged On
<input type="radio"/> Number Times Took Quiz	<input type="radio"/> Participation Cmpt100 Online Out Of10
<input type="radio"/> Number Threads Started In Course Before	<input type="radio"/> Number Posts Made In Course Before
<input type="radio"/> Number Posts Made In Category Before	<input type="radio"/> Number Posts Read In Course Before
<input type="radio"/> Number Posts Read In Category Before	

**Table Headers (all possible characteristics)**

---

**Filter Results**

Default Learner roles included:  
(If no roles are selected, the default role for the course is used instead)

**Initial group of learners in the view**

**Additional Filters to add**

---

**Set Privileges**

Choose who can generate this view (unchecked and excluded roles will not be able to generate this view):

**Groups of learners who can see the view**

---

**Describe Purpose and Save** **Description of the view**

Purpose of View:

A "view" is a table you have created to capture information about your learners. An example of a view is "a list of students who have not yet logged on". The View Creation Interface helps you create this "view".

Consider the word "characteristic" to mean a quality of the learner that can be added to the view (for example: userid, name, or number times took quiz).

Also consider the word "filter" to mean a way to limit the rows of learners being returned in the view. For example, you may filter learners out of your view who have already logged on when creating a view of students who had not yet logged on.

Please answer the following questions about the usability of the View Creation Interface:

I used the view creation screen:

☐ Frequently ☐ Occasionally ☐ Seldom ☐ Never

I was able to effectively create views for my course using the view creation screen.

☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

It was easy to use the view creation screen. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I can quickly find the characteristics I want on the view creation screen. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I can easily add a filter to my view on the view creation screen. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

The process to create a view is straightforward (the next step is obvious). ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

## List of Views Interface

The List of Views interface is accessed in the Edit Course page of iHelp Courses (using the "View Selection" link). Using this interface, you can see the results of a view after you have created it by clicking on the name of the view. The view is "generated" because it is up to date with the most recent information from your course. A sample of this screen is shown below:

Views for Course CMPT 100 Winter 2007:

View name	View Modified Date	Times Generated	Expected Time	Created by	Who can generate	Edit
<a href="#">Do the ONLINE students prefer the lecture video or the written material for XHTML?</a>	12 Feb 2007 02:40 PM	3	1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">Participation for ONLINE students</a>	05 Jan 2007 11:42 AM	22	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">time spent on network videos (inclass)</a>	19 Jan 2007 10:06 PM	5	2 seconds	Jim Greer	* all\CMPT\CMPT Faculty	<a href="#">Edit</a>
<a href="#">Who has watched the Evolution of Computers video?</a>	12 Jan 2007 09:25 AM	3	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">Which students are currently online?</a>	22 Jan 2007 03:17 PM	7	4 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">When should I log on to see someone online?</a>	05 Jan 2007 02:07 PM	5	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">who inclass saw videos module 2</a>	19 Jan 2007 09:52 PM	1	4 seconds	Jim Greer	* all\CMPT\CMPT Faculty	<a href="#">Edit</a>
<a href="#">Do the INCLASS students prefer the lecture video or the written material for XHTML?</a>	12 Feb 2007 02:41 PM	2	2 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">Which ONLINE students are currently online looking at material?</a>	22 Jan 2007 02:53 PM	4	2 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">What time of the day do learners log on?</a>	08 Jan 2007 09:40 AM	3	5 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">Is there anyone in SECTION 04 who is falling behind in the material?</a>	12 Feb 2007 02:44 PM	2	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">Last log in times and time spent on course for ONLINE students</a>	04 Jan 2007 08:33 AM	59	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">What is my participation mark?</a>	05 Jan 2007 01:54 PM	231	less than 1 second	Collene Hansen	* all\CMPT\CMPT 100 102 Online Students	<a href="#">Edit</a>
<a href="#">Which INCLASS students are using the material and how much?</a>	08 Jan 2007 09:50 AM	9	3 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">Contact information for ONLINE students who have not yet logged on</a>	04 Jan 2007 08:32 AM	10	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">viewed web and hci video</a>	21 Jan 2007 07:42 PM	6	less than 1 second	Jim Greer	* all\CMPT\CMPT Faculty	<a href="#">Edit</a>
<a href="#">How am I doing in the course compared to other ONLINE students?</a>	05 Jan 2007 01:52 PM	154	4 seconds	Collene Hansen	* all\CMPT\CMPT 100 102 Online Students	<a href="#">Edit</a>
<a href="#">Contact information for ONLINE students</a>	25 Jan 2007 09:03 AM	6	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">time spent on hci and web videos</a>	19 Jan 2007 10:17 PM	15	2 seconds	Jim Greer	* all\CMPT\CMPT Faculty	<a href="#">Edit</a>

Please answer the following questions about the usability of the List of Views Interface:

I used the list of views screen: ☐ Frequently ☐ Occasionally ☐ Seldom ☐ Never

It was easy to use the list of views screen. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I can quickly find what information I want on the list of views screen. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I am able to quickly find out what views the learners in my course are able to display. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

### View Interface

When you click on the name of a view on the list of views screen (eg. "Do the online students prefer the lecture video or the written material for XHTML"), the view screen is displayed to show you the results of that view. In other words, the system answers the question "Do the online students preer the lecture video or the written material for XHTML"). An example of the view screen is below:

Do the ONLINE students prefer the lecture video or the written material for XHTML?

Time Spent On Content And Children(XHTML and CSS)	Time Spent On Content And Children(Lecture Video: XHTML)
3 hours 55 minutes 18 seconds	35 minutes 8 seconds
8 hours 38 minutes 37 seconds	0 seconds
2 minutes 9 seconds	8 seconds
6 hours 16 minutes 10 seconds	17 seconds
3 hours 15 minutes 39 seconds	4 seconds
4 hours 50 minutes 10 seconds	2 seconds
52 minutes 36 seconds	12 seconds
3 hours 44 minutes 48 seconds	41 minutes 51 seconds
2 hours 10 minutes 57 seconds	4 seconds
6 hours 27 minutes 1 second	0 seconds
8 hours 46 minutes 40 seconds	33 seconds
19 hours 4 minutes 55 seconds	0 seconds
5 hours 51 minutes 48 seconds	0 seconds
3 hours 24 minutes 41 seconds	9 seconds
10 hours 39 minutes 49 seconds	5 seconds
5 hours 57 minutes 49 seconds	3 seconds

16 learners in table

View generated in 8 seconds

Please answer the following questions about the usability of the View Interface:

- I used the results screen: ☐ Frequently ☐ Occasionally ☐ Seldom ☐ Never
- It was easy to use the view results screen. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree
- I can quickly find what information I want on the view results screen. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree
- The view is displayed quickly after I select it. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

### General Usability

Consider the above images as examples of your interaction with the Compare Progress feature of iHelp Courses, and please answer the following questions about the usability of the Compare Progress feature as a whole.

List the most negative aspect(s) of the Compare Progress feature:

1.
2.
3.

List the most positive aspect(s) of the Compare Progress feature:

1.
2.
3.

### Effect on your Teaching

Please answer the following questions about the affect of the Compare Progress feature on your teaching strategies for your course.

- I am able to find out valuable information ☐ Strongly ☐ Agree ☐ Disagree ☐ Strongly

about learners in my course using the Compare Progress feature.

Agree

Disagree

I understand the information the Compare Progress feature provides.

☐ Strongly Agree

☐ Agree

☐ Disagree

☐ Strongly Disagree

☐ To automatically assign marks based on characteristics in the views

☐ To allow students to compare their progress to one another

☐ To identify students who are falling behind in the course

☐ To monitor the usage of course content

☐ To determine what time/day the learners log on to adjust my schedule

☐ To determine the effectiveness of the course content

☐ To monitor how well students are doing in the course

☐ To monitor participation

☐ To see what information the views provided (out of curiosity about the feature)

☐ Other (specify):

I have used the Compare Progress feature for the following reasons (check all that apply):

In order to effectively view information about learners in my course, the following characteristics are (important, may be useful, or unnecessary to me as an instructor"

	Important	May be useful, but not required	I do not need to know this to compare learners
Email	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
NSID	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Name	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Whether they are currently online	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
All their quiz marks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Their highest quiz mark	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

The number of  
postings they've  
made in the  
discussion  
forums

☐
☐
☐

The number of  
postings they've  
read in the  
discussion  
forums

☐
☐
☐

The last page  
they accessed in  
the course

☐
☐
☐

Their  
participation in  
the course

☐
☐
☐

The last time  
they accessed a  
page in the  
course

☐
☐
☐

The last time  
they logged on

☐
☐
☐

The pages  
they've viewed  
in the course

☐
☐
☐

The time they've  
spent on the  
course

☐
☐
☐

Other (specify)

☐
☐
☐

The views I generate  
fulfill the original  
purposes I had in  
mind when creating  
the views.

☐ Strongly  
Agree

☐ Agree

☐ Disagree

☐ Strongly  
Disagree

I encountered a  
situation where the  
Compare Progress  
feature did not  
support the view I was  
trying to create.

☐ Yes

☐ No

If you answered yes to  
the previous question,

describe the view you were trying to create.

The information contained in the views is inaccurate or misleading.

☐ Strongly Agree    ☐ Agree    ☐ Disagree    ☐ Strongly Disagree

If you chose strongly agree or agree in the previous question, why did you find the information inaccurate or misleading (check all that apply):

- ☐ The data itself is incorrect (eg. The learner more time on a page than the view displayed)
- ☐ The view as a whole is incorrect (eg. I believe the participation view was not indicative of participation in the class)
- ☐ The comparison is incorrect (eg. I believe some students performed better than their classmates, but the comparison did not indicate this)
- ☐ Other (specify):

I have made one or more views I have created available to my students.

☐ Yes    ☐ No

If you answered yes to the previous question, why did you choose to make the views available? (check all that apply)

- ☐ To allow students to view marks automatically generated by the system
- ☐ To allow students to see their progress in the course
- ☐ To encourage students who are falling behind to catch up to their classmates
- ☐ To allow students to compare themselves to one another (to see if they are the top learner, average, etc.)
- ☐ To encourage participation in the course by showing students how active their peers are
- ☐ To show students what course content their peers found to be the most important
- ☐ To show students what type of information I can see about them
- ☐ Other (specify):

If you answered no to the previous question, why did you choose to NOT make the

- ☐ I don't know how to make the views available
- ☐ I am concerned about the privacy of the learners in the course

views available?  
(check all that apply)

- ☐ I feel the information on the views isn't useful to the learners
- ☐ I feel the information on the views is inaccurate
- ☐ I feel the information on the views isn't needed for my teaching strategy in the course
- ☐ I feel the type of information on the views should only be available to instructors
- ☐ Other (specify):

Before using the Compare Progress feature, I felt I the learners were performing \_\_\_\_\_ in my course.

- ☐ Above average
- ☐ Average
- ☐ Below average
- ☐ I had no idea

After using the Compare Progress feature, I feel the learners are performing \_\_\_\_\_ in my course.

- ☐ Above average
- ☐ Average
- ☐ Below average
- ☐ I have no idea

Information on the views has changed my behaviour or teaching strategy for my course.

- ☐ Yes
- ☐ No

If you answered yes to the previous question, what have you changed?

Overall, the Compare Progress feature has all the functions and capabilities I expect it to have.

- ☐ Strongly Agree
- ☐ Agree
- ☐ Disagree
- ☐ Strongly Disagree

## Privacy

This section of the questionnaire asks for your opinion of the privacy protection measures of the Compare Progress feature.

Recall that the word "characteristic" for this questionnaire means a quality of the learner that can be added to a view (for example: userid, name, or number times too quiz). Also recall that the word "filter" means a way to limit the learners being



returned in the view. For example, you may filter out of your view learners who have already logged on when creating a view of learners who have not yet logged on.

There are several privacy measures used by the system to protect the privacy of learners:

- The first is the ability to set a characteristic to ALLOWED or BLOCKED for different groups of learners. For example, a learner in your class may be BLOCKED from seeing the characteristic name in the view, but your or your TA may be ALLOWED to see the characteristic name in the view.
- The second privacy measure is the ability to set a filter to ALLOWED or BLOCKED for different groups of learner. For example, you may have a filter created to limit the list of users based on their names. Again, a learner in your class may be BLOCKED from using the filter, in which case the filter would not be applied in the view for the learner, but you as an instructor are ALLOWED to run the filter to limit the learners in the view.
- Using the "Privileges" Interface (see below), the levels for each of the characteristics and filters can be set to ALLOWED or BLOCKED.

Set Data Privileges for Course ColleneTesting2

Method	all\Security	all\LCMS\CMPT 100 Online Advanced	all\devel	all\LCMS\CMPT 100 In Class	all\LCMS\CMPT 102 Guest	all\all admin
Has Role	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Userid	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked
Email	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked
First Name	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked
Last Name	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked
Recently Logged On	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked
Last Quiz Questions And Answers	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked
Last Content Accessed Exit Time	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
First Content Accessed Enter Time	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
First Content Accessed Exit Time	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
All Content Accessed Dates In Course Since	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
All Content Titles Viewed In Course	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Access Times In Last Week	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Access Times Since	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Preference To Run From Cd	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Name	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked

Set Filter Privileges for Course ColleneTesting2

Method	all\Security	all\LCMS\CMPT 100 Online Advanced	all\devel	all\LCMS\CMPT 100 In Class	all\LCMS\CMPT 102 Guest	all\all admin
Has Role	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked
Userid	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked
First Name	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Last Name	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Recently Logged On	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Viewed	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Number Times Took Quiz	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Viewer Of Page	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Logged On Since	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Accessed Course Recently	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed
Accessed Course Since	Allowed	Allowed	Allowed	Allowed	Allowed	Allowed

Set Privileges Reset

Please answer the following questions about privacy and the Compare Progress feature:

I am concerned about privacy on the web.	<input type="checkbox"/> Strongly Agree	<input type="checkbox"/> Agree	<input type="checkbox"/> Disagree	<input type="checkbox"/> Strongly Disagree
I was aware before taking this questionnaire that the system provided privacy protection for learners.	<input type="checkbox"/> Yes	<input type="checkbox"/> No		
Learner privacy is adequately protected by the system.	<input type="checkbox"/> Strongly Agree	<input type="checkbox"/> Agree	<input type="checkbox"/> Disagree	<input type="checkbox"/> Strongly Disagree
I receive useful information from the system, even with privacy protection in place.	<input type="checkbox"/> Strongly Agree	<input type="checkbox"/> Agree	<input type="checkbox"/> Disagree	<input type="checkbox"/> Strongly Disagree
Learners receive useful information from the system, even with privacy protection in place.	<input type="checkbox"/> Strongly Agree	<input type="checkbox"/> Agree	<input type="checkbox"/> Disagree	<input type="checkbox"/> Strongly Disagree
I changed the privacy level of one or more of the characteristics from ALLOWED to BLOCKED before releasing a view.	<input type="checkbox"/> Yes	<input type="checkbox"/> No		
If you answered yes to the previous question, why did you change the privacy level?	<input type="text"/>			
I changed the privacy level of one or more of the characteristics from BLOCKED to ALLOWED before releasing a view.	<input type="checkbox"/> Yes	<input type="checkbox"/> No		
If you answered yes to the previous question, why did you change the privacy level?	<input type="text"/>			
I believe an individual learner may be identified using information generated by the views I created.	<input type="checkbox"/> Strongly Agree	<input type="checkbox"/> Agree	<input type="checkbox"/> Disagree	<input type="checkbox"/> Strongly Disagree
Learners should be allowed to create views using the Compare Progress feature.	<input type="checkbox"/> Strongly Agree	<input type="checkbox"/> Agree	<input type="checkbox"/> Disagree	<input type="checkbox"/> Strongly Disagree

### Additional Features

Several new extensions to the Compare Progress feature are being considered to help instructors view additional information, or help instructors use the current information more effectively. How would you rate the importance of the following possible extensions to increase the effectiveness of the Compare Progress feature?

Adding a wizard to import an existing view from another course.

☐ Absolutely Required
 ☐ Nice to have
 ☐ Not necessary
 ☐ Would decrease the usefulness of the feature

Creating your own custom characteristics based on other characteristics available in the Compare Progress feature. For example, calculating your own participation mark from other characteristics such as number of chat messages, page views, etc.

☐ Absolutely Required
 ☐ Nice to have
 ☐ Not necessary
 ☐ Would decrease the usefulness of the feature

Adding aggregate data to the columns such as average, maximum, and minimum values.

☐ Absolutely Required
 ☐ Nice to have
 ☐ Not necessary
 ☐ Would decrease the usefulness of the feature

Adding styles or colors to the columns and rows to indicate additional information such as the top learner, learners to watch, etc.

☐ Absolutely Required
 ☐ Nice to have
 ☐ Not necessary
 ☐ Would decrease the usefulness of the feature

Adding new view types other than the table, such as a tree view, pie chart view, or a time-elapsed view.

☐ Absolutely Required
 ☐ Nice to have
 ☐ Not necessary
 ☐ Would decrease the usefulness of the feature

Suggestion for another feature (optional):

☐ Absolutely Required
 ☐ Nice to have
 ☐ Not necessary
 ☐ Would decrease the usefulness of the feature

Suggestion for another feature (optional):

☐ Absolutely Required
 ☐ Nice to have
 ☐ Not necessary
 ☐ Would decrease the usefulness of the feature

## **Comments**

Thank you for your time and comments! If you have any further comments about the Compare Progress feature that were not addressed in the questionnaire, please use the space below.

## APPENDIX E: LEARNER CONSENT FORM AND SURVEY

Learner Consent Form

Department of Computer Science

University of Saskatchewan  
Informed Consent Form

Research  
Project:

Active Open Learner Modelling for  
Reflection (AOLMR)

Investigators:

Dr. Gordon McCalla, Professor,  
Department of Computer Science (966-  
4902), mccalla@cs.usask.ca

Collene Hansen, Instructor (CMPT 100  
Online), Department of Computer Science  
(966-8647), collene@cs.usask.ca

We are investigating the usability and effect of the Compare Progress feature of the iHelp Courses system (as part of the Active Open Learner Modelling for Reflection project, AOLMR). We would like your consent to participate in this study. This consent form should give you the basic idea of what the research is about and what your participation will involve. If you would like more detail about something mentioned here, or information is not included here, please contact one of the study investigators listed above. Please take the time to read this form carefully and to understand any accompanying information.

This study will observe the class interactions and system usage of the Compare Progress feature of the iHelp Courses system (as part of the Active Open Learner Modelling for Reflection project, AOLMR). The Compare Progress feature has been designed as a study tool to help online students reflect on their progress in the course

by comparing their progress and activities in the course compared to their peers. The main goal of the study will be to gain insight into the usability of the Compare Progress feature and its impact on reflection in an actual class. CMPT 100 Online was targeted as a candidate for this tool because of the class discussion, participation, and online delivery aspects of the class. The Investigators are confident that the class will also benefit from the tool.

The data collected from this study will be used in articles for publication in journals and conference proceedings. As one way of thanking you for your time, we will be pleased to make available to you a summary of the results of this study once they have been compiled. This summary will outline the research and discuss our findings and recommendations.

All of the information we collect from you (data logged by the computer, observations made by the experimenters, and your questionnaire responses) will be stored so that your name, student number, nsid, or email address is not associated with it. Any write-ups of the data will not include any information that can be linked directly to you. The research materials will be stored with complete security throughout the entire investigation.

**By signing this form, you give permission to the Investigators to use examples taken from your interaction with the Compare Progress feature of iHelp Courses. None of your identifying information will be shared with anyone outside of the Investigators. The data in fact will also appear anonymously to the Investigators -- it will not contain information that would be able to link it to you. If you do not sign, we will only be able to use statistical measures of system usage. In any event, the information collected will not be used until after your final course grade has been submitted. None of the information collected will influence your grade in this or other courses.**

Your continued participation should be as informed as your initial consent, so you should feel free to ask for clarification or new information throughout your participation. If you have further questions concerning matters related to this research, please contact one of the investigators listed above.

By choosing the "Yes, I would like to participate" option below, you indicate that you have understood to your satisfaction the information regarding participation in the research project and agree to participate. In no way does this waive your legal rights nor release the investigators, sponsors, or involved institutions from their legal and professional responsibilities. If you have further questions about this study or your rights as a participant, please contact:

- Dr. Gordon McCalla, Professor Dept. Computer Science (306) 966-4902  
mccalla@cs.usask.ca
- Office of Research Ethics University of Saskatchewan (306) 966-2084

**You are free to opt out at any time and need only inform an Investigator.**

If you would like a copy of this consent form please print it through your browser, or contact one of the investigators listed above. This research has the ethical approval of the Office of Research Services at the University of Saskatchewan.

Date: April 2nd, 2007

**Signed:**  
(please enter your full name)

**Email:**  
(for possible contact)

- ☒ Yes, I would like to participate
- ☐ No, I don't want to take part

### iHelp Courses Compare Progress - Learner Survey

Thank you for your participation in this questionnaire. The responses you provide will be used to evaluate the effectiveness of the Compare Progress feature in iHelp Courses and modify the feature for future course offerings. Please limit your responses to comments about the Compare Progress feature only (to be described in the questionnaire), and not about the iHelp applications in general.

There are several parts to this questionnaire:

- The first part of the questionnaire asks about the interface of the Compare Progress feature. These questions will ask your opinion on the ease of use of the interface.
- The second part of the questionnaire will ask you about how the Compare Progress feature affected your learning in your course (for example, if you changed your behavior as a result of viewing of the information).
- The third part of the questionnaire asks about privacy in the Compare Progress feature (for example, whether you feel your privacy is protected by the system, what level of privacy protection you feel is appropriate, etc).
- The last part of the questionnaire asks your opinion about additional features in the Compare Progress feature, and leaves space for your comments.

### Compare Progress Usability

#### List of Views Interface

The list of views available for you in your course is displayed when you click on the "Compare Progress" action item in the Actions menu in iHelp Courses.

**Actions:**

-  [Update Preferences](#)
-  [Compare Progress](#)
-  [Logout](#)

**Views for Course CMPT 100 Winter 2007:**

<a href="#">View name</a>	<a href="#">View Modified Date</a>	<a href="#">Times Generated</a>	<a href="#">Expected Time</a>
<a href="#">How am I doing in the course compared to other ONLINE students?</a>	05 Jan 2007 01:52 PM	155	4 seconds
<a href="#">What is my participation mark?</a>	05 Jan 2007 01:54 PM	231	less than 1 second

Please answer the following questions about the usability of the list of views screen of the Compare Progress Interface (an example of the list of views screen is shown in the image above):

I used the list of views screen:

- ☐ Frequently   ☐ Occasionally   ☐ Seldom   ☐ Never

It was easy to use the list of views screen.

- ☐ Strongly Agree   ☐ Agree   ☐ Disagree   ☐ Strongly Disagree

I can quickly find what information I want on the list of views screen.

- ☐ Strongly Agree   ☐ Agree   ☐ Disagree   ☐ Strongly Disagree

I can quickly find what views I was able to generate for the course.

- ☐ Strongly Agree   ☐ Agree   ☐ Disagree   ☐ Strongly Disagree

**View Interface**

When you click on the name of a view in the list of views screen (for example "How am I doing?"), the view screen is displayed to show you the results of that view (in other words, to answer the question "How am I doing?"). An example of the view results screen is shown below.



How am I doing in the course compared to other ONLINE students?

Userid()	Participation Cmp100 Online Out DT10(CMPT 100 Winter 2007)	Percentage Posts Read In Course(CMPT 100 Winter 2007)	Time Spent On Content And Children(CMPT 100 Winter 2007)	Last Content Title Accessed In Course(CMPT 100 Winter 2007)
-BLOCKED-	9.0	78.99544	3 days 10 hours 46 minutes 54 seconds	CMPT 100 Winter 2007\Course Content\Artificial Intelligence\Introduction and Notable People\The Turing Test.
-BLOCKED-	3.0	12.100456	2 days 4 hours 31 minutes 29 seconds	CMPT 100 Winter 2007\Course Content\Artificial Intelligence
-BLOCKED-	8.0	48.74429	1 day 13 hours 52 minutes 15 seconds	CMPT 100 Winter 2007\Assignments\CMPT 100 Online Assignments\A9: Artificial Intelligence
-BLOCKED-	9.0	80.93607	2 days 14 hours 36 minutes 30 seconds	CMPT 100 Winter 2007\Course Content\Artificial Intelligence
-BLOCKED-	9.0	74.77169	2 days 7 hours 59 minutes 39 seconds	CMPT 100 Winter 2007\Assignments\CMPT 100 Online Assignments\A9: Artificial Intelligence
-BLOCKED-	3.0	11.872147	17 hours 14 minutes 6 seconds	CMPT 100 Winter 2007\Assignments\CMPT 100 Online Assignments\A9: Artificial Intelligence
-BLOCKED-	9.0	80.82192	1 day 4 hours 6 minutes 22 seconds	CMPT 100 Winter 2007\Course Content\Artificial Intelligence
-BLOCKED-	8.0	39.269405	3 days 2 hours 44 minutes 42 seconds	CMPT 100 Winter 2007\Course Content\Information Technology Security
-BLOCKED-	6.0	75.45662	1 day 10 hours 4 minutes 26 seconds	CMPT 100 Winter 2007\Course Content\Artificial Intelligence\Methods\Artificial Neural Networks
-BLOCKED-	4.0	16.666668	2 days 19 hours 5 minutes 4 seconds	CMPT 100 Winter 2007\CMPT 100 Online Lounge
-BLOCKED-	2.0	29.3379	22 hours 59 minutes 35 seconds	CMPT 100 Winter 2007\Assignments\CMPT 100 Online Assignments\A10: Social Implications
-BLOCKED-	9.0	79.56621	3 days 3 hours 38 minutes 25 seconds	CMPT 100 Winter 2007\Course Content\Artificial Intelligence
-BLOCKED-	3.0	34.246574	1 day 17 hours 29 minutes 21 seconds	CMPT 100 Winter 2007\Course Content\Programming Languages and JavaScript\Introduction to JavaScript
-BLOCKED-	4.0	10.159818	2 days 21 hours 22 minutes 36 seconds	CMPT 100 Winter 2007\Assignments\CMPT 100 Online Assignments\A9: Artificial Intelligence
-BLOCKED-	9.0	81.39269	2 days 16 hours 21 minutes 27 seconds	CMPT 100 Winter 2007\Assignments\CMPT 100 Online Assignments\A9: Artificial Intelligence
-BLOCKED-	8.0	81.39269	1 day 12 hours 35 minutes 42 seconds	CMPT 100 Winter 2007\Course Content\Computers In Society\Lecture Notes: 3 to a page

16 learners in table

View generated in less than 1 second

Please answer the following questions about the usability of the view results screen:

I used the view  
results screen:

☐ Frequently ☐ Occasionally ☐ Seldom ☐ Never

It was easy to use the  
view results screen.

☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I can quickly find  
what information I  
want on the view  
results screen.

☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I know what  
information is about  
me on the view  
results screen.

☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

The view results  
screen is displayed  
quickly after I select  
it.

☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

## General Usability

Consider the above images as examples of your interaction with the Compare Progress feature in iHelp Courses, and please answer the following questions about the usability of the Compare Progress feature as a whole:

List the most negative aspect(s) of the Compare Progress feature:

1.
2.
3.

List the most positive aspect(s) of the Compare Progress feature:

1.
2.
3.

## Effect on your Learning

Please answer the following questions about the effect of the Compare Progress feature on your learning in your course.

The word "characteristic" for this questionnaire means a quality about you or a peer in your course that may be displayed on a view (for example: userid, participation mark, number of postings read in a category, etc).

I am able to find valuable information about my progress in my course using the Compare Progress feature.

☐ Strongly Agree    ☐ Agree    ☐ Disagree    ☐ Strongly Disagree

The view names are self-explanatory (meaning I know what information the view will display).

☐ Strongly Agree    ☐ Agree    ☐ Disagree    ☐ Strongly Disagree

I understand the information the Compare Progress feature provides.

☐ Strongly Agree    ☐ Agree    ☐ Disagree    ☐ Strongly Disagree

In order to be able to effectively compare myself to other learners in my course, the following characteristics are (important, may be useful, or unnecessary):

	<b>Important</b>	<b>May be useful, but not required</b>	<b>I do not need to know this to compare myself to other learners</b>
Email	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
NSID	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Name	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Whether they are currently online	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
All their quiz marks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Their highest quiz mark	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The number of postings they've made in the discussion forums	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The number of postings they've read in the discussion forums	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The last page they accessed in the course	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Their participation in the course	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The last time they accessed a page in the course	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The last time they logged on	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The pages they've viewed in the course	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

The time they've spent on the course

☐
☐
☐

Other (specify)

☐
☐
☐


I encountered a situation where the Compare Progress feature did not provide information that I wanted to know.

☐

Yes

☐

No

If you answered yes to the previous question, describe the information you wanted to know:

☐

To make sure that I am keeping up with material in the course

☐

To see if I am the top student in the course

☐

To see if I am above average in the course

☐

To see my mark which is automatically generated by the system

☐

To affect my mark which is automatically generated by the system

☐

To see how I compare to my classmates in the course

☐

To see what information someone else could find out about me

☐

To identify course content that I should spend more time on

☐

To find out what course content my peers have considered important

☐

To see what information the views provided (out of curiosity about the feature)

☐

Other (specify):

The information contained in the views is inaccurate or misleading

☐

Strongly Agree

☐

Agree

☐

Disagree

☐

Strongly Disagree

If you chose strongly agree or agree in the previous question, why did you find the information inaccurate or misleading (check all that apply):

- ☐ The data itself is incorrect (eg. I spent more time on a page than the view displayed)
- ☐ The view as a whole is incorrect (eg. I believe the participation view was not indicative of participation in the class)
- ☐ The comparison is incorrect (eg. I believe I performed better than my classmates, but the comparison did not indicate this)
- ☐ Other (specify):

Information contained in the views changed my behavior in my course.

- |  |   |  |   |
|--|---|--|---|
| <input type="checkbox"/> Yes, the information had a significant affect | <input type="checkbox"/> Yes, the information had a little affect | <input type="checkbox"/> No, the information had no affect | <input type="checkbox"/> I did not view the information |
|--|---|--|---|

If you answered yes to the previous question, what have you done differently in your course as a result of seeing the information in the views?

Before using the Compare Progress feature, I felt I was performing \_\_\_\_\_ in my course.

- |  |                                  |  |  |
|--|----------------------------------|--|--|
| <input type="checkbox"/> Above average | <input type="checkbox"/> Average | <input type="checkbox"/> Below average | <input type="checkbox"/> I had no idea |
|--|----------------------------------|--|--|

After using the Compare Progress feature, I feel I am performing \_\_\_\_\_ in my course.

- |  |                                  |  |   |
|--|----------------------------------|--|---|
| <input type="checkbox"/> Above average | <input type="checkbox"/> Average | <input type="checkbox"/> Below average | <input type="checkbox"/> I have no idea |
|--|----------------------------------|--|---|

Overall, the Compare Progress feature has all the functions and capabilities I expect it to have.

- |   |                                |                                   |  |
|---|--------------------------------|-----------------------------------|--|
| <input type="checkbox"/> Strongly Agree | <input type="checkbox"/> Agree | <input type="checkbox"/> Disagree | <input type="checkbox"/> Strongly Disagree |
|---|--------------------------------|-----------------------------------|--|

## Privacy

This section of the questionnaire asks for your opinion of the privacy protection measures of the Compare Progress tools.

The word "characteristic" for this questionnaire means a quality about you or a peer in your course that may be displayed on a view (for example: userid, participation mark, number of postings read in a category, etc).

There are several privacy measures used by the system to protect the privacy of learners:

- Instructors can set a characteristic to ALLOWED or BLOCKED for different groups of people. For example, you peers may be BLOCKED from seeing names, but your instructor is still able to see your name.
- You can always see all information about yourself on a view.
- Instructors can't view **all** information about you in the system (for instance, they can not see which Aliases belong to you, they can not see what college you belong to, and other personal information), but they can view such characteristics as how long you've spent on topics, whether you've read a particular page, how many posts you've created and read in the discussion forums, and quiz marks.

Please answer the following questions about privacy and the Compare Progress feature:

I am concerned about privacy on the web. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I was aware before taking this questionnaire that the system provided privacy protection for learners ☐ Yes ☐ No

Learner privacy is protected at an appropriate level by the system. ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I received useful information from the system, even with privacy protection in ☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

place

I believe it may be possible to to specifically identify another learner using information on the views I had access to in my course.

☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I was able to specifically identify another learner using information provided by the Compare Progress feature.

☐ Yes, and I did identify another learner ☐ Yes, I thought of a way to specifically identify a learner, but I did not use it to actually identify someone ☐ Maybe, it may be possible, but I'm not sure ☐ No, it is not possible to identify another learner

I would feel comfortable if I was allowed to create views using the Compare Progress features (meaning I could request what characteristics were displayed on the view).

☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I would feel comfortable if my classmates were allowed to create views using the Compare Progress feature.

☐ Strongly Agree ☐ Agree ☐ Disagree ☐ Strongly Disagree

I feel comfortable revealing the following characteristics about myself to...

	All my classmates	Only those who I choose (friends, group members, etc)	My instructor
Email	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
NSID	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Name	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Whether I'm currently online	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
All my quiz marks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
My highest quiz mark	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The number of postings I've made in the discussion forums	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The number of postings I've read in the discussion forums	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The last page I accessed in the course	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
My participation in the course	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The last time I accessed a page in the course	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The last time I logged on	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
The pages I've viewed in the course	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

What else could be done to protect learner privacy?

## Additional Features

Several new extensions to the Compare Progress feature are being considered to help users view the information more effectively. How would you rate the importance of the following possible extensions to increase the effectiveness of the Compare Progress feature?



Adding aggregate data to the columns (such as average, minimum and maximum values).	<input type="checkbox"/> Absolutely Required	<input type="checkbox"/> Nice to have	<input type="checkbox"/> Not necessary	<input type="checkbox"/> Would decrease the usefulness of the feature
Adding styles or colors to the columns and rows to indicate additional information such as the top learner, average learner, etc.	<input type="checkbox"/> Absolutely Required	<input type="checkbox"/> Nice to have	<input type="checkbox"/> Not necessary	<input type="checkbox"/> Would decrease the usefulness of the feature
Creating new view types other than the table, such as a tree view, a pie chart view, or a time-elapsd view.	<input type="checkbox"/> Absolutely Required	<input type="checkbox"/> Nice to have	<input type="checkbox"/> Not necessary	<input type="checkbox"/> Would decrease the usefulness of the feature
Suggestion for another feature (optional):	<input type="checkbox"/> Absolutely Required	<input type="checkbox"/> Nice to have	<input type="checkbox"/> Not necessary	<input type="checkbox"/> Would decrease the usefulness of the feature
Suggestion for another feature (optional):	<input type="checkbox"/> Absolutely Required	<input type="checkbox"/> Nice to have	<input type="checkbox"/> Not necessary	<input type="checkbox"/> Would decrease the usefulness of the feature

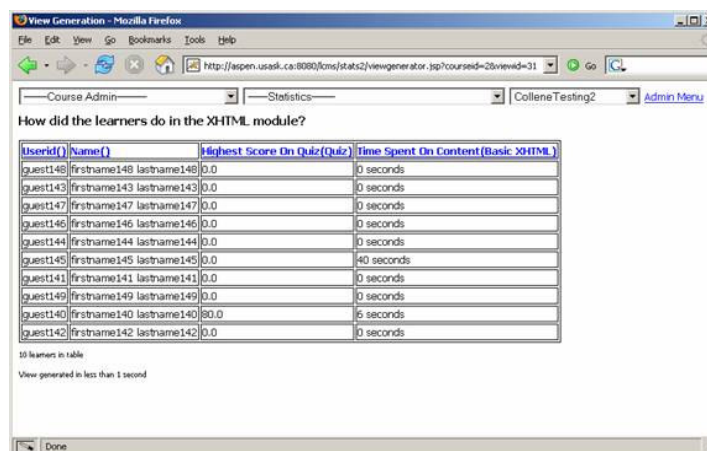
### Comments

Thank you for your time and comments! If you have any further comments about the Compare Progress feature that were not addressed in the questionnaire, please use the space below.

## APPENDIX F: QUERY TOOL USER'S GUIDE

It is possible to view the activities and learner trends in your course using the statistics tools in iHelp Courses (also called the Compare Progress Feature). For example, you may need to identify “how did the learners do in this module?”, “are learners viewing the online videos accompanying the course”, and “what is the participation in the course”. Instructors may view this information, and may open the information to the learners in their course for comparison, if they choose to do so.

Currently, there is one "view" you can create – a "table view". A "table view" of learners is a table with rows of learners, and columns of learner characteristics, such as their user id, first name, last name, score on a quiz, etc. Figure F-1 is an example of a table view. In the example, the instructor would like to know "how did the learners do in the XHTML module". This purpose may mean different things to different instructors, but for this instructor he/she wanted to see the highest score on the quiz in the module, and the time the learners spent on the material.



View Generation - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://aspen.usask.ca:8080/learn/stats2/viewgenerator.jsp?courseid=28&viewid=31

Course Admin Statistics ColleneTesting2 Admin Menu

How did the learners do in the XHTML module?

Userid()	Name()	Highest Score On Quiz(Quiz)	Time Spent On Content(Basic XHTML)
quest148	firstname148 lastname148	0.0	0 seconds
quest143	firstname143 lastname143	0.0	0 seconds
quest147	firstname147 lastname147	0.0	0 seconds
quest146	firstname146 lastname146	0.0	0 seconds
quest144	firstname144 lastname144	0.0	0 seconds
quest145	firstname145 lastname145	0.0	40 seconds
quest141	firstname141 lastname141	0.0	0 seconds
quest149	firstname149 lastname149	0.0	0 seconds
quest140	firstname140 lastname140	80.0	6 seconds
quest142	firstname142 lastname142	0.0	0 seconds

10 learners in table

View generated in less than 1 second

Done

Figure F-1: Example of a table view in iHelp Courses

## Creating a View

A table view can be created using the “Create Stats Table View for Learners” option in the Edit courses menu of iHelp Courses (Figure F-2).



Figure F-2: Edit courses menu of iHelp Courses

There are 4 areas on this page: Model Table View Creation, Filter Results, Set Privileges, and Describe Purpose and Save (Figure F-3).

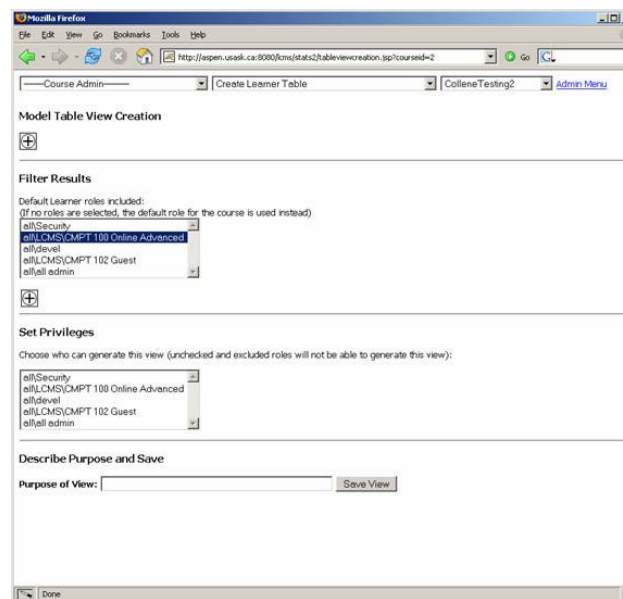



Figure F-3: View creation interface

## Model (Table View) Creation

In this area, an instructor specifies which columns (characteristics) to have in the table. Click the  button to add a column. The default column that comes up is the "userid" column (Figure F-4). All the characteristics available to the instructor are selectable with radio buttons below the new column. The characteristics are descriptive by title. Add a desired characteristic to the view by selecting the radio button.

userid  

<input type="radio"/> Has Role	<input checked="" type="radio"/> Userid
<input type="radio"/> Email	<input type="radio"/> First Name
<input type="radio"/> Last Name	<input type="radio"/> Viewed
<input type="radio"/> Times User Took Quiz	<input type="radio"/> Last Time User Took Quiz
<input type="radio"/> First Time User Took Quiz	<input type="radio"/> Scores On Quiz
<input type="radio"/> Lowest Score On Quiz	<input type="radio"/> Average Score On Quiz
<input type="radio"/> Highest Score On Quiz	<input type="radio"/> Quiz Marks In Course
<input type="radio"/> Quizzes Not Completed In Course	<input type="radio"/> Quizzes Completed In Course
<input type="radio"/> Number Quizzes Completed In Course	<input type="radio"/> Number Times Took Quiz
<input type="radio"/> Number Threads Started In Category Before	<input type="radio"/> Number Threads Started In Category
<input type="radio"/> Number Threads Started In Course Before	<input type="radio"/> Number Threads Started In Course
<input type="radio"/> Number Posts Made In Course Before	<input type="radio"/> Number Posts Made In Course
<input type="radio"/> Number Posts Made In Category Before	<input type="radio"/> Number Posts Made In Category
<input type="radio"/> Average Posting Reply Lag Time In Course	<input type="radio"/> Average Posting Reply Lag Time In Category
<input type="radio"/> Average Posting Read Lag Time In Category	<input type="radio"/> Average Posting Read Lag Time In Course
<input type="radio"/> Percentage Posts Read In Course	<input type="radio"/> Percentage Posts Read In Category
<input type="radio"/> Participation Cmp100 Online Out Of10	<input type="radio"/> Participation Cmp100 Online
<input type="radio"/> Number Posts Read In Course Before	<input type="radio"/> Number Posts Read In Course
<input type="radio"/> Number Posts Read In Category Before	<input type="radio"/> Number Posts Read In Category
<input type="radio"/> Last Post Date In Category	<input type="radio"/> Last Read Date In Category
<input type="radio"/> Last Category Read Date In Course	<input type="radio"/> Number Messages In Channel Compared To Total
<input type="radio"/> Number Messages In Course	<input type="radio"/> Number Messages In Channel
<input type="radio"/> Last Message Date In Channel	<input type="radio"/> Last Message Date In Course
<input type="radio"/> First Login Date	<input type="radio"/> Last Login Date
<input type="radio"/> Number Logins	<input type="radio"/> Recently Logged On
<input type="radio"/> Number Logins In Past Week	<input type="radio"/> Number Logins Since
<input type="radio"/> Usual Login Times Of The Day	<input type="radio"/> Usual Login Days Of The Week
<input type="radio"/> All Logins	<input type="radio"/> All Logins Since
<input type="radio"/> Last Content Title Accessed In Course	<input type="radio"/> Last Content Accessed Date
<input type="radio"/> Last Content Accessed Date In Course	<input type="radio"/> Time Spent On Content And Children
<input type="radio"/> Time Spent On Content	<input type="radio"/> Number Times Accessed Content
<input type="radio"/> All Content Accessed Dates In Course	<input type="radio"/> Last Content Accessed Enter Time
<input type="radio"/> Last Content Accessed Exit Time	<input type="radio"/> First Content Accessed Enter Time
<input type="radio"/> First Content Accessed Exit Time	<input type="radio"/> All Content Accessed Dates In Course Since
<input type="radio"/> All Content Titles Viewed In Course	<input type="radio"/> All Quiz Questions And Answers
<input type="radio"/> Access Times In Last Week	<input type="radio"/> Access Times Since
<input type="radio"/> Last Quiz Questions And Answers	<input type="radio"/> Preference To Run From Cd
<input type="radio"/> Name	

Figure F-4: Adding characteristics to a view

Some characteristics may require additional parameters to be entered, in which case a new box will be displayed on the right hand side of the window with the allowable options for the characteristic. For example, choosing "Highest Score on Quiz" displays a select box with all the quizzes in the course, and the instructor then chooses the particular quiz he/she is interested in (as in Figure F-5).

Model Table View Creation

highestScoreOnQuiz(ColleneTesting2\Computational Linguistics\Quiz) X ↔ +


<input type="radio"/> Has Role	<input type="radio"/> Userid
<input type="radio"/> Email	<input type="radio"/> First Name
<input type="radio"/> Last Name	<input type="radio"/> Viewed
<input type="radio"/> Times User Took Quiz	<input type="radio"/> Last Time User Took Quiz
<input type="radio"/> First Time User Took Quiz	<input type="radio"/> Scores On Quiz
<input type="radio"/> Lowest Score On Quiz	<input type="radio"/> Average Score On Quiz
<input checked="" type="radio"/> Highest Score On Quiz	<input type="radio"/> Quiz Marks In Course
<input type="radio"/> Quizzes Not Completed In Course	<input type="radio"/> Quizzes Completed In Course
<input type="radio"/> Number Quizzes Completed In Course	<input type="radio"/> Number Times Took Quiz
<input type="radio"/> Number Threads Started In Category	<input type="radio"/> Number Threads Started In Course
<input type="radio"/> Number Posts Made In Course	<input type="radio"/> Number Posts Made In Category
<input type="radio"/> Average Posting Reply Lag Time In Course	<input type="radio"/> Average Posting Reply Lag Time In Category
<input type="radio"/> Average Posting Read Lag Time In Category	<input type="radio"/> Average Posting Read Lag Time In Course
<input type="radio"/> Percentage Posts Read In Course	<input type="radio"/> Percentage Posts Read In Category
<input type="radio"/> Participation Cmp100 Online Out Of10	<input type="radio"/> Participation Cmp100 Online
<input type="radio"/> Number Posts Read In Course	<input type="radio"/> Number Posts Read In Category
<input type="radio"/> Last Post Date In Category	<input type="radio"/> Last Read Date In Category
<input type="radio"/> Last Category Read Date In Course	<input type="radio"/> Number Messages In Channel Compared To Total
<input type="radio"/> Number Messages In Course	<input type="radio"/> Number Messages In Channel
<input type="radio"/> Last Message Date In Channel	<input type="radio"/> Last Message Date In Course
<input type="radio"/> First Login Date	<input type="radio"/> Last Login Date
<input type="radio"/> Number Logins	<input type="radio"/> Recently Logged On
<input type="radio"/> Number Logins In Past Week	<input type="radio"/> Number Logins Since
<input type="radio"/> Usual Login Times Of The Day	<input type="radio"/> Usual Login Days Of The Week
<input type="radio"/> All Logins	<input type="radio"/> All Logins Since
<input type="radio"/> Last Content Title Accessed In Course	<input type="radio"/> Last Content Accessed Date
<input type="radio"/> Last Content Accessed Date In Course	<input type="radio"/> Time Spent On Content And Children
<input type="radio"/> Time Spent On Content	<input type="radio"/> Number Times Accessed Content
<input type="radio"/> All Content Accessed Dates In Course	<input type="radio"/> Last Content Accessed Enter Time
<input type="radio"/> Last Content Accessed Exit Time	<input type="radio"/> First Content Accessed Enter Time
<input type="radio"/> First Content Accessed Exit Time	<input type="radio"/> All Content Accessed Dates In Course Since
<input type="radio"/> All Content Titles Viewed In Course	<input type="radio"/> All Quiz Questions And Answers
<input type="radio"/> Access Times In Last Week	<input type="radio"/> Access Times Since
<input type="radio"/> Last Quiz Questions And Answers	<input type="radio"/> Preference To Run From Cd
<input type="radio"/> Name	

Parameters for this data type:

Quiz:

Figure F-5: Adding extra information to the view using parameters

Notice as well that the currently selected column along the top (identified by the yellow background) changed when the "Highest Score on Quiz" option was selected. Selecting a different quiz from the drop down box will also update the description in the current column.





To add another column to the table, click the  button again. A new column will be added, and will become the selected column with the yellow background. Again, the default "userid" is the initial characteristic selected here, but this can be changed by choosing one of the radio buttons from the list (Figure F-6).

Model Table View Creation

highestScoreOnQuiz(ColleneTesting2\Computational Linguistics\Quiz)  **userid**  

<input type="radio"/> Has Role	<input checked="" type="radio"/> Userid
<input type="radio"/> Email	<input type="radio"/> First Name
<input type="radio"/> Last Name	<input type="radio"/> Viewed
<input type="radio"/> Times User Took Quiz	<input type="radio"/> Last Time User Took Quiz
<input type="radio"/> First Time User Took Quiz	<input type="radio"/> Scores On Quiz
<input type="radio"/> Lowest Score On Quiz	<input type="radio"/> Average Score On Quiz
<input type="radio"/> Highest Score On Quiz	<input type="radio"/> Quiz Marks In Course
<input type="radio"/> Quizzes Not Completed In Course	<input type="radio"/> Quizzes Completed In Course
<input type="radio"/> Number Quizzes Completed In Course	<input type="radio"/> Number Times Took Quiz
<input type="radio"/> Number Threads Started In Category	<input type="radio"/> Number Threads Started In Course
<input type="radio"/> Number Posts Made In Course	<input type="radio"/> Number Posts Made In Category
<input type="radio"/> Average Posting Reply Lag Time In Course	<input type="radio"/> Average Posting Reply Lag Time In Category
<input type="radio"/> Average Posting Read Lag Time In Category	<input type="radio"/> Average Posting Read Lag Time In Course
<input type="radio"/> Percentage Posts Read In Course	<input type="radio"/> Percentage Posts Read In Category
<input type="radio"/> Participation Cmpt100 Online Out Of10	<input type="radio"/> Participation Cmpt100 Online
<input type="radio"/> Number Posts Read In Course	<input type="radio"/> Number Posts Read In Category
<input type="radio"/> Last Post Date In Category	<input type="radio"/> Last Read Date In Category
<input type="radio"/> Last Category Read Date In Course	<input type="radio"/> Number Messages In Channel Compared To Total
<input type="radio"/> Number Messages In Course	<input type="radio"/> Number Messages In Channel
<input type="radio"/> Last Message Date In Channel	<input type="radio"/> Last Message Date In Course
<input type="radio"/> First Login Date	<input type="radio"/> Last Login Date
<input type="radio"/> Number Logins	<input type="radio"/> Recently Logged On
<input type="radio"/> Number Logins In Past Week	<input type="radio"/> Number Logins Since
<input type="radio"/> Usual Login Times Of The Day	<input type="radio"/> Usual Login Days Of The Week
<input type="radio"/> First Login Date	<input type="radio"/> Last Login Date
<input type="radio"/> Last Content Title Accessed In Course	<input type="radio"/> Last Content Accessed Date
<input type="radio"/> Last Content Accessed Date In Course	<input type="radio"/> Time Spent On Content And Children
<input type="radio"/> Time Spent On Content	<input type="radio"/> Number Times Accessed Content
<input type="radio"/> All Content Accessed Dates In Course	<input type="radio"/> Last Content Accessed Enter Time
<input type="radio"/> Last Content Accessed Exit Time	<input type="radio"/> First Content Accessed Enter Time
<input type="radio"/> First Content Accessed Exit Time	<input type="radio"/> All Content Accessed Dates In Course Since
<input type="radio"/> All Content Titles Viewed In Course	<input type="radio"/> All Quiz Questions And Answers
<input type="radio"/> Access Times In Last Week	<input type="radio"/> Access Times Since
<input type="radio"/> Last Quiz Questions And Answers	<input type="radio"/> Preference To Run From Cd
<input type="radio"/> Name	


Figure F-6: Adding a second characteristic to the view

Each column in the table has a few icons within it () to modify the table in case a different ordering is desired, or to remove the column entirely. The  will remove the column from the table.  moves the column one place to the left, and  moves the column one place to the right.

## Filtering Results

The instructor may not want to see all the learners in the course in the model being created. If so, he/she may filter out learners based on characteristics of the learners. For example, an instructor may only want to see contact information for learners who have not yet logged on to the course, so he/she can send an email to the student. The instructor may also want to find out who is falling behind in the material, and may base an assumption of falling behind on whether the learner has viewed a particular page in the course.

The view starts with a particular group of learners – the roles identified as students for the course. For example, in the test course there are several roles: Security, CMPT 100 Online Advanced, devel, CMPT 100 In Class, etc. An instructor can choose one or more of these roles as a maximum set of learners by choosing the role from the select box (Figure F-7). For example, the instructor may choose to create the view starting with students in the course, or may create the view starting with the teaching assistants in the course. In the example in Figure F-7, the instructor has chosen the role CMPT 100 Online Advanced as the initial group of learners.

Filters are then applied one at a time, starting with this initial group, filtering out learners as filters are applied. To add a filter, click the  button (Figure F-7). The default filter is “Recently Logged On”, but the instructor may change the filter to another if desired by selecting the radio button for the corresponding filter. Just as in the case of the characteristics/columns, if the filter requires additional parameters, a box will appear below the filters asking for the parameters.






## Filter Results


Default Learner roles included:

(If no roles are selected, the default role for the course is used instead)

all\Security
all\LCMS\CMPT 100 Online Advanced
all\devel
all\LCMS\CMPT 100 In Class
all\LCMS\CMPT 102 Guest
all\all admin

Filter In: Recently Logged On  




---

Filter In 
---

<input type="radio"/> Has Role	<input type="radio"/> Userid	<input type="radio"/> First Name
<input type="radio"/> Last Name	<input type="radio"/> Viewed	<input type="radio"/> Number Times Took Quiz
<input checked="" type="radio"/> Recently Logged On	<input type="radio"/> Viewer Of Page	<input type="radio"/> Logged On Since
<input type="radio"/> Accessed Course Recently	<input type="radio"/> Accessed Course Since	

Figure F-7: Adding a filter

The filter may be applied to “Filter In” or “Filter Out”, depending on the desired behavior. For example, if the instructor wants to include only those learners who have logged on recently, he/she would select “Filter In”. If the instructor wants to exclude those learners who have recently logged on, he/she would select “Filter Out”. In Figure F-7, the instructor has chosen to “Filter In” learners who have recently logged on.

To continue adding filters, click the  button. Filters can be moved up or down (so as to increase or decrease the precedence of the filters) using the  icons on each filter.

## Setting Privileges

Instructors may allow the new view to be revealed to other learners in the community. For example, a model calculating participation and a model showing



the learners how they compare to their classmates would be beneficial to students in a course. Teaching assistants may require additional information such as contact information or a class list so they can contact learners.

To give a group of learners access to the view (so they are also able to view it), choose the role from the set privileges select box on the view creation interface (Figure F-8). The instructor can give more than one role access to the view by selecting multiple options in the select box. If the instructor does not want anyone else to run this view, he/she can simply leave the roles as unselected.

### Set Privileges

Choose who can generate this view (unchecked and excluded roles will not be able to generate this view):

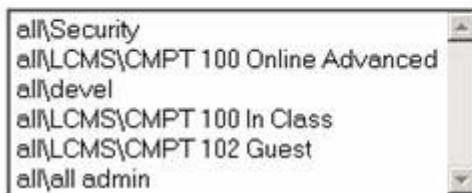


Figure F-8: Giving other users access to a view

### Describe Purpose and Save

At the bottom of the view creation interface is an area where the instructor specifies a purpose for the view, or a description to what the view will display (Figure F-9). The purpose should give the view a descriptive name so others are aware of what the view will create. For example, an instructor may wish to choose the purpose “What is my participation as compared to others in the course” if he/she wants the students to generate such a model for that purpose. An instructor may also give the purpose “Who is falling behind in my section?”. The name of the purpose provides a description of what the instructor plans to do with the characteristics in the model, so the name should be very descriptive.

## Describe Purpose and Save

Purpose of View:

Figure F-9: Describe the purpose of the view and save

Once the purpose has been added, click the  button. A blue box will appear while the view is being saved: Saving view, please wait. Once the view has been saved correctly, a green box will appear: View has been saved. If an error has occurred, a red box will appear with a warning message -- in this case you should contact the iHelp Courses Administrator with details about the view you tried to create.

The view will now be saved and will be available on the list of views screen so it can be displayed.

## Listing and Generating Views

Once a view has been created, the instructor or other user able to use the view must generate it in order to see the results. A generated view will be up to date to consider the most recent activities in the course, and is calculated when the user requests it. A list of views already created can be accessed using the “View Selection” option in the Edit course menu of iHelp Courses (Figure F-10).

Views for Course ColleneTesting2:

View name	View Modified Date	Times Generated	Expected Time	Created by	Who can generate	Edit
<a href="#">testing before functions</a>	23 Jan 2007 12:55 PM	1	1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test</a>	06 Oct 2006 11:51 AM	0	0 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">blah!</a>	05 Jan 2007 04:29 PM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test quizzes</a>	06 Oct 2006 11:49 AM	3	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test new table</a>	06 Oct 2006 11:50 AM	2	2 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test since</a>	23 Jan 2007 12:00 PM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">How did the learners do in the XHTML module?</a>	06 Oct 2006 05:44 PM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test</a>	12 Feb 2007 04:09 PM	0	0 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test filter in section 02 only</a>	18 Jan 2007 05:01 PM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test 2</a>	06 Oct 2006 11:49 AM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">testing participation marks</a>	06 Oct 2006 11:50 AM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">testing TWO PARAMETERS</a>	23 Jan 2007 12:53 PM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">testing NEW TABLES</a>	06 Oct 2006 11:50 AM	2	1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test filter in section 02 only</a>	18 Jan 2007 05:00 PM	0	0 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">testing before functions 2</a>	23 Jan 2007 12:54 PM	2	1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">quiz question answers</a>	06 Oct 2006 11:51 AM	0	0 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test 1</a>	06 Oct 2006 11:50 AM	2	less than 1 second	Collene Hansen	* all\level * all\LCMS\CMPT 102 Guest	<a href="#">Edit</a>
<a href="#">test run messages in course</a>	06 Oct 2006 11:50 AM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">blah!</a>	05 Jan 2007 04:29 PM	0	0 seconds	Collene Hansen	No one	<a href="#">Edit</a>

Figure F-10: List of views for a test course

The view name is the purpose for which the view was created, set when the view was initially saved. Other information is included in this table as well, including when the view was modified, how popular it is, how long the view will take to generate, who created the view, and who can generate the view.


To view/generate one of the views, click the view name. A new window with the learner model will appear with the results of the view as in Figure F-11 (it may take a few seconds for the page to be created).

Userid()	Name()	Highest Score On Quiz(Quiz)	Time Spent On Content(Basic XHTML)
guest148	firstname148 lastname148	0.0	0 seconds
guest143	firstname143 lastname143	0.0	0 seconds
guest147	firstname147 lastname147	0.0	0 seconds
guest146	firstname146 lastname146	0.0	0 seconds
guest144	firstname144 lastname144	0.0	0 seconds
guest145	firstname145 lastname145	0.0	40 seconds
guest141	firstname141 lastname141	0.0	0 seconds
guest149	firstname149 lastname149	0.0	0 seconds
guest140	firstname140 lastname140	80.0	6 seconds
guest142	firstname142 lastname142	0.0	0 seconds

10 learners in table  
View generated in less than 1 second

Done

Figure F-11: A generated view

Alternatively, you may also view a list of learner models while viewing course content in iHelp Courses. Click the  [Compare Progress](#) link on the actions menu on the left while viewing a page in the course (Figure F-12). The views for the course will show up on the right, minus some of the course editing information such as who created the view and who can generate the view (Figure F-13).

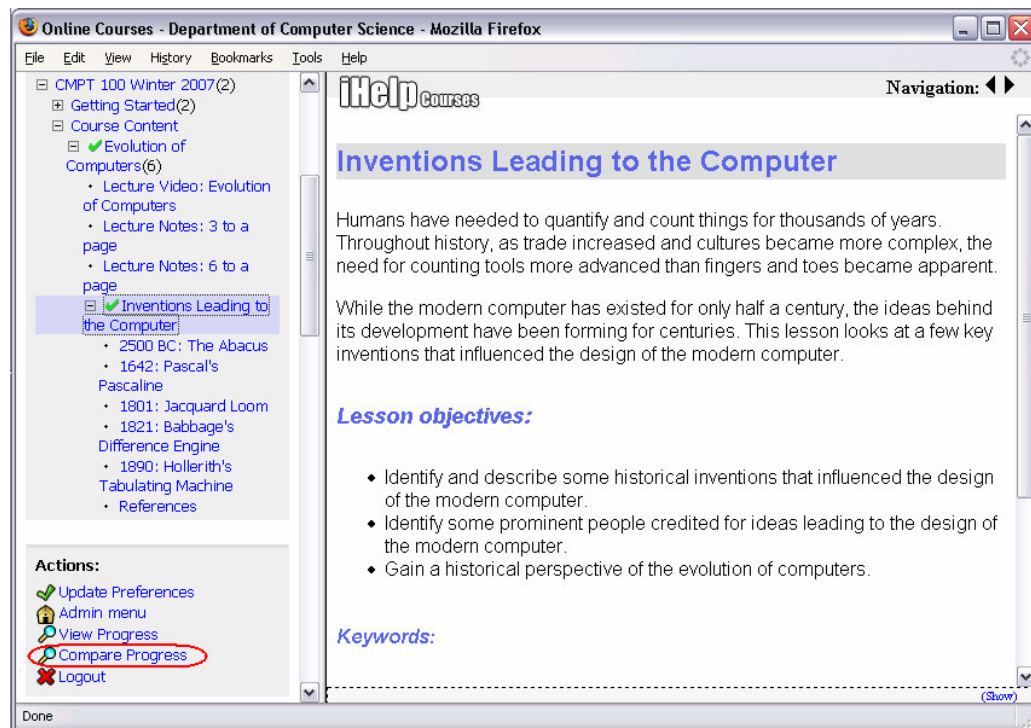


Figure F-12: Actions menu of iHelp Courses

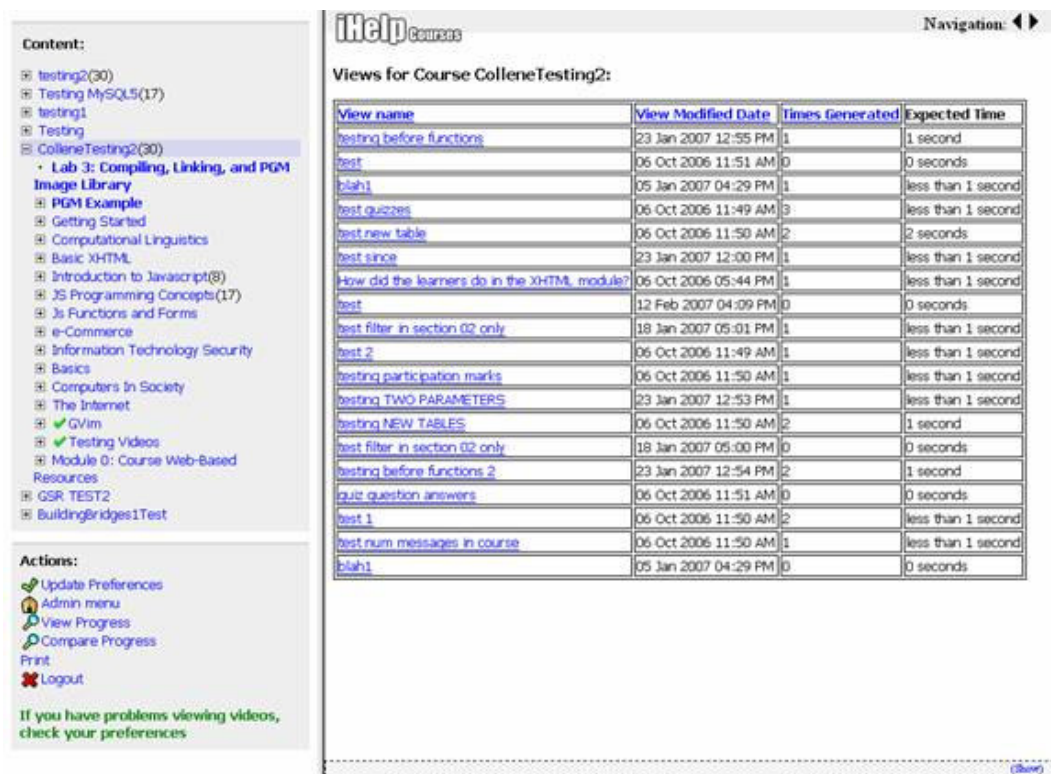


Figure F-13: Listing views using the iHelp Courses actions menu

This Compare Progress link is how the learners will interact with the views in the course – any views the instructor has made available to the students in the course will show up in this list.

## Editing a View

An instructor can edit a previously created view, creating an entirely new view which is similar to the old one, or refining the current one. To edit the view, click the Edit link next to the view name while viewing a list of views in iHelp Courses through the Edit Course menu (Figure F-14).

Views for Course ColleneTesting2:






View name	View Modified Date	Times Generated	Expected Time	Created by	Who can generate	Edit
<a href="#">testing before functions</a>	23 Jan 2007 12:55 PM	1	1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test</a>	06 Oct 2006 11:51 AM	0	0 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">blah1</a>	05 Jan 2007 04:29 PM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test quizzes</a>	06 Oct 2006 11:49 AM	3	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test new table</a>	06 Oct 2006 11:50 AM	2	2 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test since</a>	23 Jan 2007 12:00 PM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">How did the learners do in the XHTML module?</a>	06 Oct 2006 05:44 PM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test</a>	12 Feb 2007 04:09 PM	0	0 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test filter in section 02 only</a>	18 Jan 2007 05:01 PM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test 2</a>	06 Oct 2006 11:49 AM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">testing participation marks</a>	06 Oct 2006 11:50 AM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">testing TWO PARAMETERS</a>	23 Jan 2007 12:53 PM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">testing NEW TABLES</a>	06 Oct 2006 11:50 AM	2	1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test filter in section 02 only</a>	18 Jan 2007 05:00 PM	0	0 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">testing before functions 2</a>	23 Jan 2007 12:54 PM	2	1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">quit question answers</a>	06 Oct 2006 11:51 AM	0	0 seconds	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">test 1</a>	06 Oct 2006 11:50 AM	2	less than 1 second	Collene Hansen	* all/devel * all/LCMS/CMP1 102 Guest	<a href="#">Edit</a>
<a href="#">test num messages in course</a>	06 Oct 2006 11:50 AM	1	less than 1 second	Collene Hansen	No one	<a href="#">Edit</a>
<a href="#">blah1</a>	05 Jan 2007 04:29 PM	0	0 seconds	Collene Hansen	No one	<a href="#">Edit</a>

Figure F-14: Listing views for editing

The current view will be displayed. Edit the view similar to how the view was initially created (Figure F-15). Select a characteristic by clicking the name of it in the view.




## Model Table View Creation

[userid](#)  [name](#)  [highestScoreOnQuiz\(Quiz\)](#)  [timeSpentOnContent\(Basic XHTML\)](#)  

## Filter Results

Default Learner roles included:


(If no roles are selected, the default role for the course is used instead)

  
all\Security  
all\LCMS\CMPT 100 Online Advanced  
all\devel  
all\LCMS\CMPT 100 In Class  
all\LCMS\CMPT 102 Guest  
all\all admin



## Set Privileges

Choose who can generate this view (unchecked and excluded roles will not be able to generate this view):

  
all\Security  
all\LCMS\CMPT 100 Online Advanced  
all\devel  
all\LCMS\CMPT 100 In Class  
all\LCMS\CMPT 102 Guest  
all\all admin

## Describe Purpose and Save

**Purpose of View:**

- ☒ Overwrite existing view with this one  
☐ Keep existing view, create new view

Figure F-15: Editing a view

There are two options at the bottom of the Edit screen – “Overwrite existing view with this one” or “Keep existing view, create new view”. The first option does a “Save” function -- it deletes the old view, and replaces it with this new edited view. The second option does a “Save As” function – the old view is kept as it originally was, and the edited view is added to the list of views. This “Save As” functionality can be useful when the instructor is creating several views with the same columns

but different parameters (i.e. a view for the same information but for different modules).

## Setting Privacy

Some of the characteristics in the learner models may be sensitive information that an instructor (or learners) may not want others to know. For example, names or other identifying information may be private in some courses, but public in others. An instructor can set privileges for each group to Allow or Block each characteristic.

Privileges can be set using the “Set Data Privileges” option in the Edit course menu of iHelp Courses. Note that it may take a few seconds for this window to appear while the current permissions are being loaded (Figure F-16).

Set Data Privileges for Course ColleneTesting2


Method	all Security	all LCMS\CMPT 100 Online Advanced	all devel	all LCMS\CMPT 100 In Class	all LCMS\CMPT 102 Guest	all admin
Has Role	Allowed ▾	Allowed ▾	Allowed ▾	Allowed ▾	Allowed ▾	Allowed ▾
Userid	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾
Email	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾
First Name	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾
Last Name	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾	Blocked ▾
Viewed	Allowed ▾	Allowed ▾	Allowed ▾	Allowed ▾	Allowed ▾	Allowed ▾

Figure F-16: Setting privacy

Notice that some groups may be permitted to see some of the characteristics while others may not – this is up to the instructor for the course. Identifying information (such as name and userid) is blocked by default by the system, without privileges being set.

Filters can also be blocked or allowed. Some filters may allow individual learners to be identified (for example, userid or currently logged on), so are blocked by default by the system. The instructor may choose to Allow or Block the filters based on roles in the course. In the case of a blocked filter, the learner’s view will appear as if the filter had not been applied (so all of the learners will be displayed), and a

message at the bottom of the window tells the learner a filter had not been applied based on his/her permissions.

Once the desired levels have been set for the characteristics and filters, click the  button to save the privileges. Note that it may take a few seconds for the privileges to be set.